OPTIMALISASI ALGORITMA *XGBOOST* MENGGUNAKAN *HYPERPARAMETER TUNING* DAN *MULTIPLE PREPROCESSING* UNTUK PREDIKSI HARGA MOBIL BEKAS

TESIS

Diajukan sebagai salah satu syarat untuk memperoleh gelar

Magister Ilmu Komputer (M.Kom)

DANIATI UKI EKA SAPUTRI

14002372

Program Studi Ilmu Komputer (S2)

Universitas Nusa Mandiri

Jakarta

2021

SURAT PERNYATAAN ORISINALITAS DAN BEBAS PLAGIARISME

Yang bertanda tangan di bawah ini :

Nama	: Daniati Uki Eka Saputri
NIM	: 14002372
Program Studi	: Ilmu Komputer
Jenjang	: Strata Dua (S2)
Konsentrasi	: Data Mining

Dengan ini menyatakan bahwa tesis yang telah saya buat dengan judul: "Optimalisasi Algoritma *XGBoost* menggunakan *Hyperparameter Tuning* dan *Multiple Preprocessing* untuk Prediksi Harga Mobil Bekas" adalah hasil karya sendiri, dan semua sumber baik yang kutip maupun yang dirujuk telah saya nyatakan dengan benar dan tesis belum pernah diterbitkan atau dipublikasikan dimanapun dan dalam bentuk apapun.

Demikianlah surat pernyataan ini saya buat dengan sebenar-benarnya Apabila dikemudian hari ternyata saya memberikan keterangan palsu dan atau ada pihak lain yang mengklaim bahwa tesis yang telah saya buat adalah hasil karya milik seseorang atau badan tertentu, saya bersedia diproses baik secara pidana maupun perdata dan kelulusan saya dari Program Studi Ilmu Komputer (S2) Universitas Nusa Mandiri dicabut/dibatalkan.

Jakarta, 04 Agustus 2021
Yang Menyatakan,

Daniati Uki Eka Saputri

HALAMAN PERSETUJUAN DAN PENGESAHAN TESIS

Tesis ini diajukan oleh:

Nama	:	Daniati Uki Eka Saputri
NIM	:	14002372
Program Studi	:	Ilmu Komputer
Jenjang	:	Strata Dua (S2)
Konsentrasi	:	*Data Mining*
Judul Tesis	:	Optimalisasi Algoritma XGBoost Menggunakan Hyperparameter Tuning Dan Multiple Preprocessing Untuk Prediksi Harga Mobil Bekas

Telah dipertahankan pada periode 2021-1 dihadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh Magister Ilmu Komputer (M.Kom) pada Program Studi Ilmu Komputer (S2) Universitas Nusa Mandiri.

Jakarta, 12 Agustus 2021

PEMBIMBING TESIS

Pembimbing I	:	Dr. Muhammad Haris, M.Eng

D E W A N P E N G U J I

Penguji I	:	Dr. Windu Gata, M.Kom
Penguji II	:	Dr. Rifki Sadikin, M.Kom
Penguji III / Pembimbing I	:	Dr. Muhammad Haris, M.Eng

UNIVERSITAS NUSA MANDIRI	LEMBAR BIMBINGAN TESIS
	UNIVERSITAS NUSA MANDIRI

Nama : Daniati Uki Eka Saputri
NIM : 14002372
Program Studi : Ilmu Komputer
Jenjang : Strata Dua (S2)
Konsentrasi : Data Mining
Judul Tesis : Optimalisasi Algoritma *XGBoost* menggunakan *Hyperparameter Tuning* dan *Multiple Preprocessing* untuk Prediksi Harga Mobil Bekas

NO	Tanggal Bimbingan	Pokok Bahasan	Paraf Dosen Pembimbing
1	07/04/2021	Mengumpulkan PPT dan proposal	
2	17/04/2021	Arahan tema, data dan algoritma	
3	24/04/2021	Laporan progres penelitian	
4	30/05/2021	Laporan progress bimbingan sebelumnya	
5	23/06/2021	Pembahasan koding yang error	
6	06/07/2021	Mengumpulkan draft tesis Bab 1-5	
7	17/07/2021	Revisi laporan tesis Bab 1-5	
8	02/08/2021	ACC keseluruhan	

Catatan untuk dosen pembimbing
Bimbingan Tesis
- Dimulai pada tanggal : 07 April 2021
- Diakhiri pada tanggal : 02 Agustus 2021
- Jumlah pertemuan bimbingan : 8 (Delapan) kali bimbingan

Disetujui Oleh,
Dosen Pembimbing

(Dr. Muhammad Haris, M.Eng)

KATA PENGANTAR

Puji syukur alhamdullillah, penulis panjatkan kehadirat Allah, SWT, yang telah melimpahkan rahmat dan karunia-Nya, sehingga pada akhirnya penulis dapat menyelesaikan laporan tesis ini tepat pada waktunya. Dimana laporan tesis ini penulis sajikan dalam bentuk buku yang sederhana.

Adapun judul tesis, yang penulis ambil sebagai berikut "Optimalisasi Algoritma *XGBoost* menggunakan *Hyperparameter Tuning* dan *Multiple Preprocessing* untuk Prediksi Harga Mobil Bekas".

Tujuan penulisan laporan tesis ini dibuat sebagai salah satu untuk mendapatkan gelar Magister Ilmu Komputer (M.Kom) pada Program Studi Ilmu Komputer (S2) Universitas Nusa Mandiri.

Laporan Tesis ini diambil berdasarkan hasil penelitian atau riset mengenai prediksi harga mobil bekas. Penulis juga lakukan mencari dan menganalisa berbagai macam sumber referensi, baik dalam bentuk jurnal ilmiah, buku-buku literatur, *internet*, dll yang terkait dengan pembahasan pada laporan tesis ini.

Penulis menyadari bahwa tanpa bimbingan dan dukungan dari semua pihak dalam pembuatan laporan tesis ini, maka penulis tidak dapat menyelesaikan laporan tesis ini tepat pada waktunya. Untuk itu ijinkanlah penulis kesempatan ini untuk mengucapkan.terimakasih kepada :

1. Rektor Universitas Nusa Mandiri Ibu Dr. Dwiza Riana, S.Si, MM, M.Kom
2. Wakil Rektor Universitas Nusa Mandiri Ibu Nita Merlina, M.Kom
3. Kepala Program Studi Ilmu Komputer Bapak Dr. Hilman F Pardede, ST, M.EICT bersama Sek.Prodi Ilmu Komputer Ibu Eni Heni Hermaliani, MM. M.Kom.
4. Bapak Dr. Muhammad Haris, M.Eng selaku pembimbing tesis yang telah menyediakan waktu, pikiran dan tenaga dalam membimbing penulis dalam menyelesaikan tesis ini.
5. Orang tua tercinta yang telah memberikan dukungan material dan moral kepada penulis.
6. Seluruh Dosen Program Studi Ilmu Komputer (S2) Universitas Nusa Mandiri yang telah memberikan pelajaran yang berarti bagi penulis selama menempuh studi.

7. Seluruh staf di lingkungan Universitas Nusa Mandiri yang telah melayani penulis dengan baik selama kuliah.
8. dan lain-lain

Serta semua pihak yang terlalu banyak untuk penulis sebutkan satu persatu sehingga terwujudnya penulisan laporan tesis ini. Penulis menyadari bahwa penulisan laporan tesis ini masih jauh sekali dari sempurna, untuk itu penulis mohon kritik dan saran yang bersifat membangun demi kesempurnaan penulisan karya ilmiah yang penulis hasilkan untuk yang akan datang.

Akhir kata semoga laporan tesis ini dapat bermanfaat bagi penulis khususnya dan bagi para pembaca yang berminat pada umumnya.

Jakarta, 04 Agustus 2021

Daniati Uki Eka Saputri
Penulis

SURAT PERNYATAAN PERSETUJUAN PUBLIKASI KARYA ILMIAH

UNTUK KEPENTINGAN AKADEMIS

Yang bertanda tangan di bawah ini, saya :

Nama	: Daniati Uki Eka Saputri
NIM	: 14002372
Program Studi	: Ilmu Komputer
Jenjang	: Strata Dua (S2)
Konsentrasi	: Data Mining
Jenis	: Tesis

Demi pengembangan ilmu pengetahuan, dengan ini menyetujui untuk memberikan ijin kepada pihak Program Studi Ilmu Komputer (S2) Universitas Nusa Mandiri **Hak Bebas Royalti Non-Eksklusif *(Non-exclusive Royalti-Free Right)*** atas karya ilmiah kami yang berjudul : "Optimalisasi Algoritma *XGBoost* menggunakan *Hyperparameter Tuning* dan *Multiple Preprocessing* untuk Prediksi Harga Mobil Bekas" beserta perangkat yang diperlukan (apabila ada).

Dengan **Hak Bebas Royalti Non-Eksklusif** ini pihak Universitas Nusa Mandiri berhak menyimpan, mengalih-media atau *bentuk*-kan, mengelolaannya dalam pangkalan data *(database)*, mendistribusikannya dan menampilkan atau mempublikasikannya di *internet* atau media lain untuk kepentingan akademis tanpa perlu meminta ijin dari kami selama tetap mencantumkan nama kami sebagai penulis/pencipta karya ilmiah tersebut.

Saya bersedia untuk menanggung secara pribadi, tanpa melibatkan pihak Universitas Nusa Mandiri, segala bentuk tuntutan hukum yang timbul atas pelanggaran Hak Cipta dalam karya ilmiah saya ini .

Demikian pernyataan ini saya buat dengan sebenarnya.

Jakarta, 04 Agustus 2021

Yang menyatakan,

Daniati Uki Eka Saputri

ABSTRAK

Nama	: Daniati Uki Eka Saputri
NIM	: 14002372
Program Studi	: Ilmu Komputer
Jenjang	: Strata Dua (S2)
Konsentrasi	: Data Mining
Judul	:"Optimalisasi Algoritma *XGBoost* menggunakan *Hyperparameter Tuning* dan *Multiple Preprocessing* untuk Prediksi Harga Mobil Bekas"

Adanya pandemi *covid-19* secara tidak langsung meningkatkan minat masyarakat terhadap kendaraan yang nyaman,aman serta harga yang terjangkau, dan mobil bekas menjadi pilihan terbaik. semakin tingginya minat akan mobil bekas, berdampak pada terciptanya peluang usaha *showroom* mobil bekas. Dalam menentukan harga jual mobil bekas, dibutuhkan pengetahuan ahli. Tujuan dari penelitian ini untuk mendapatkan model prediksi harga yang akurat sehingga dapat membantu *showroom* bersaing harga dengan penjual lain. Pihak *showroom* akan memahami fitur-fitur penting yang berpengaruh terhadap kenaikan dan penurunan harga. Selain itu, dapat membantu pembeli dalam mengambil keputusan memilih mobil bekas sesuai dengan kondisi dan harganya. Metode yang digunakan untuk penelitian yaitu *XGBoost* dengan melakukan optimalisasi *hyperparameter tuning* dan *multiple preprocessing* data. Hasil penelitian menghasilkan *MAE*= 0,218, dan *R2*= 0.89%, hasil tersebut lebih baik dari penelitian sebelumnya dengan *Gradient Boosted Regression tree* menghasilkan *MAE*=0,28 dan dengan *random forest regression* menghasilkan *R2*=83,63%. Dengan *Hyperparameter Tuning* dan *Multiple Preprocessing* dapat meningkatkan akurasi prediksi.

Kata Kunci : Prediksi, *XGBoost, Hyperparameter, Preprocessing*

ABSTRACT

Nama	: Daniati Uki Eka Saputri
NIM	: 14002372
Program Studi	: Ilmu Komputer
Jenjang	: Strata Dua (S2)
Konsentrasi	: Data Mining
Judul	: "*Optimization XGBoost Algorithm using Hyperparameter Tuning and Multiple Preprocessing for Used Car Price Prediction*"

The existence of COVID-19 pandemic has indirectly increased public interest in comfortable, safe and affordable vehicles, and used cars are the best choice. The increasing interest in used cars has an impact on the creation of business opportunities for used car showrooms. In determining the selling price of a used car, expert knowledge is needed. The purpose of this research to get an accurate price prediction model so that it can help showrooms to compete with other sellers. showrooms will understand the important features that affect price increases and decreases. In addition, it can assist buyers in making decisions about choosing a used car according to its condition and price. The method used for this research is XGBoost by optimizing hyperparameter tuning and multiple preprocessing data. The results showed MAE = 0.218, and R2 = 0.89%, these results are better than previous studies with Gradient Boosted Regression tree producing MAE = 0.28 and random forest regression producing R2 = 83.63%. With Hyperparameter Tuning and Multiple Preprocessing can improve prediction accuracy.

***Keywords**: Prediction, XGBoost, Hyperparameter, Preprocessing*

DAFTAR ISI

Halaman

DAFTAR TABEL

Halaman

DAFTAR GAMBAR

Halaman

DAFTAR LAMPIRAN

Halaman

BAB I

PENDAHULUAN

1.1. Latar Belakang Penulisan

Penjualan kendaraan bermotor, termasuk mobil baru maupun mobil bekas pada tahun 2019 menunjukkan angka mencapai 1.030.126 unit [1]. Sedangkan di masa pandemi covid-19 ini, data penjualan mobil dari Gabungan Industri Kendaraan Bermotor Indonesia (Gaikindo) pada tahun 2020 menunjukkan penurunan sebesar 44% dari tahun sebelumnya. Pandemi covid-19 secara tidak langsung meningkatkan minat masyarakat terhadap kendaraan yang nyaman,aman serta harga yang terjangkau, dan mobil bekas menjadi pilihan terbaik [2].

CARRO yaitu sebuah platform jual beli mobil di Asia Tenggara, berhasil menunjukkan terjadinya peningkatan permintaan mobil bekas di tahun 2020 dan terus meningkat di tahun 2021 total penjualan unit mobil bekas hingga di atas 100% dibandingkan di tahun 2020 [2]. Sementara itu, semakin tingginya minat akan mobil bekas, berdampak pada terciptanya peluang usaha bagi pebisnis mobil seperti *showroom* mobil bekas. Namun, untuk tetap dapat mempertahankan eksistensi sebuah usaha mobil bekas diperlukan kecepatan dan ketepatan dalam menentukan harga jual sehingga mampu bersaing dipasaran luar[3].

Memprediksi harga mobil bekas saat ini menjadi *trend* yang popular, banyak *showroom* mobil bekas yang bersaing harga untuk mendapatkan konsumen. Dalam menentukan harga jual mobil pada iklan penjualan mobil bekas tidak dapat dilakukan begitu saja, dibutuhkan pengetahuan ahli [4]. Perlu diperhatikan fitur-fitur yang mempengaruhinya, seperti tahun mobil dibuat, model, asal pabrikan, jarak tempuhnya, tenaga, bahan bakar, keadaan fisik, transmisi yang digunakan dalam mobil yang dapat mempengaruhi harga naik turun sebuah mobil [5]. Oleh karena itu,dapat disimpulkan bahwa harga mobil dipengaruhi oleh banyak fitur. Namun, fitur-fitur tersebut masih belum cukup mendukung penentuan harga mobil bekas, sehingga perlu memiliki model prediksi harga mobil bekas secara akurat berdasarkan fitur-fitur yang ada.

Machine learning mampu melakukan proses pembelajaran dengan fakta dan pengetahuan, serta mampu menemukan pola dan aturan pada kumpulan data baru yang belum pernah dipelajari sebelumnya untuk mendapatkan pengetahuan baru [6]. Dengan data yang besar, *machine learning* mampu membangun konstruksi model inferensi yang dilakukan secara otomatis [7]. Karena kecanggihannya, *machine learning* digunakan di banyak aspek kehidupan salah satunya yaitu prediksi penjualan data yang diperoleh di masa lampau.

Tujuan dari penelitian ini yaitu mendapatkan model algoritma *machine learning* yang memberikan hasil paling akurat dalam memprediksi harga jual mobil bekas. Penelitian ini penting dilakukan karena pasar mobil bekas semakin popular dan banyak bermunculan *showroom* maupun iklan-iklan penjualan mobil bekas. Ketepatan dalam memprediksi harga mobil bekas akan membantu penjual maupun pembeli. Penjual akan lebih memahami fitur-fitur penting yang berpengaruh terhadap kenaikan ataupun penurunan harga, sehingga dapat memberikan harga yang jauh lebih baik kepada konsumen. Selain itu, model prediksi yang akurat akan membantu pembeli dalam mengambil keputusan untuk memilih mobil bekas sesuai dengan kondisi dan harganya.

Merujuk dari penelitian yang telah dilakukan oleh [8] dengan melakukan komparasi kinerja berbasis regresi yaitu *Gradient Boosted Regression tree, Random Forest, dan Multiple Linear Regression* menghasilkan kinerja terbaiknya berada pada algoritma *Gradient Boosted Regression tree* dengan *Mean Squared Error(MSE*)=0.28, Diikuti dengan *Random Fores*t sebesar *MSE*=0.35,dan *Multiple Linear Regression* sebesar *MSE*=0.55. Dari penelitian tersebut, penulis mengajukan peningkatan performa yaitu terjadi penurunan tingkat *error* ketika menggunakan algoritma *Gradient Boosted Regression tree.*

Selain itu, penelitian yang telah dilakukan oleh [9] mengenai prediksi harga mobil bekas dengan menggunakan algoritma *linear regression* dan *random forest regression* melakukan beberapa *preprocessing* data dan menggunakan sepuluh fitur mobil bekas yang digunakan untuk prediksi harga

mobil bekas. Akurasi terbaik ditemukan ketika 500 pohon keputusan digunakan untuk membangun model pada algoritma *random forest regression*. Evaluasi yang digunakan yaitu dengan *R2* skor dengan nilai Skor pelatihan adalah 95,82%, dan skor pengujian adalah 83,63%. Fitur paling relevan yang digunakan untuk prediksi adalah harga, kilometer, merek, dan jenis kendaraan dengan memfilter pencilan dan fitur yang tidak relevan dari kumpulan data

Pada kesempatan ini, peneliti menerapkan algoritma berbasis regresi yaitu *XGBoost* untuk memprediksi harga mobil bekas. Data mobil bekas tersebut memiliki distribusi data yang tidak merata sehingga terjadi banyak pencilan atau banyak *outlier* yang dapat mengganggu performa dari prediksi harga mobil bekas. Dari kasus tersebut, kemudian diterapkan transformasi log dan normalisasi data menggunkaan metode *RobustScaler*. Model yang dihasilkan kemudian akan dievaluasi dengan *Mean Square Error*, *Mean Absolute Error* & *R2 Score* untuk mengetahui keakuratan model prediksi harga mobil bekas. Hasil penelitian ini diharapkan mampu memberikan manfaat untuk penjual mobil bekas atau pemilik *showroom* mobil bekas dalam menyiapkan strategi penjualan di masa mendatang. Selain itu, diharapkan juga memberikan manfaat untuk masyarakat luas khususnya mereka yang ingin membeli mobil bekas agar dapat dengan mudah mengambil keputusan dalam menentukan mobil bekas yang sesuai dengan kondisi dan harganya.

1.2. Identifikasi Masalah

Berdasarkan latar belakang masalah tersebut, maka permasalahan dalam penelitian ini yaitu :

1. Apakah model *XGBoost* yang paling berpengaruh dalam meningkatkan performa model prediksi harga penjualan mobil bekas secara akurat?
2. Apakah proses *preprocessing* yang memberikan dampak paling efektif untuk prediksi harga penjualan mobil bekas?
3. Bagaimana tingkat *error* hasil prediksi harga penjualan mobil bekas menggunakan metrik *Mean Square Error, Mean Absolute Error* & *R2 Score*?

4. Atribut apa saja yang memiliki hubungan paling berpengaruh terhadap prediksi harga penjualan mobil bekas?

1.3. Tujuan Penelitian

Berdasarkan latar belakang masalah tersebut, maka tujuan dari penelitian ini adalah :

1. Mendapatkan model terbaik dari algoritma *XGBoost* untuk memprediksi harga penjualan mobil bekas secara akurat.
2. Mendapatkan metode *preprocessing* yang paling efektif untuk memprediksi harga penjualan mobil bekas
3. Mengetahui tingkat *error* pada hasil prediksi harga penjualan mobil bekas
4. Menganalisa hubungan antar atribut terhadap hasil prediksi harga penjualan mobil bekas

1.4. Ruang Lingkup Penelitian

Permasalahan yang diambil dalam penelitian ini dibatasi pada memprediksi harga penjualan mobil bekas dengan 20 variabel dan 371.539 *record* pada data penjualan mobil bekas *Ebay-Kleinanzeigen* yang diperoleh dari *data world* [10]. *Data world* merupakan sebuah situs *online* yang menyediakan berbagai macam data yang bersifat umum.. Penelitian menggunakan algoritma *machine learning* yaitu *XGBoost*. Selain itu, dilakukan Analisa performa algoritma dengan metode *RobustScaler* sebagai normalisasi data dan deteksi *outlier* dengan persentil serta transformasi log. Untuk metode evaluasi perfoma model menggunakan *Mean Square Error, Mean Absolute Error* & *R2 Score.*

1.5. Hipotesa

1. Parameter algoritma *XGBoost* yang tepat dan mampu menangani masalah prediksi harga penjualan mobil bekas serta dapat memberikan nilai kerugian yang minimum.
2. Metode *preprocessing* data yang tepat dan mampu menangani masalah prediksi harga penjualan mobil bekas serta dapat memberikan nilai kerugian yang minimum.

3. Tingkat error menggunakan metrik *Mean Square Error, Mean Absolute Error* dan akurasi *R2* skor akurat untuk prediksi harga mobil bekas.
4. Terdapat atribut yang berperan positif dalam meningkatkan performa model algoritma model algoritma *XGBoost*

BAB II
LANDASAN/KERANGKA PEMIKIRAN

2.1 Tinjauan Pustaka

Tinjauan Pustaka digunakan untuk menjelaskan secara teoritis yang berkaitan dengan pengertian atau definisi yang berhubungan dengan masalah yang sedang diangkat dalam penelitian untuk mendapatkan kesamaan persepsi. Tinjauan Pustaka ini berisi refrensi yang diambil dari buku, jurnal nasional dan jurnal internasional yang berhubungan dengan masalah yang sedang diangkat dalam penelitian.

2.1.1 Prediksi Harga Mobil

Prediksi dalam bahasa ilmiah merupakan perkiraan di masa yang akan datang dengan cara matematis berdasarkan pengetahuan dan pengalaman dari masa lampau serta model prediksi mampu memecahkan masalah-masalah seperti prediksi dan estimasi [11]. Prediksi atau peramalan juga disebut sebagai proses memperkirakan sesuatu yang dilakukan berdasarkan data masa lampau yang dianalisis dengan metode tertentu kemudian dipelajari dan dikaitkan dengan perjalanan waktu/*time series* yang mana hasil dari sebuah prediksi dapat dijadikan sebagai unsur dalam mengambil sebuah keputusan di masa yang akan dating [12].

Data *time series* sendiri merupakan data yang terurut berdasarkan waktu yang diperoleh secara berkala untuk melihat perkembangan dari suatu produk atau barang tersebut sehingga menghasilkan suatu gambaran estimasi atau prediksi untuk tahun-tahun kedepan [13]. Prediksi tidak harus menghasilkan jawaban pasti dari apa yang diperkirakan, namun berusaha mencari jawaban yang paling dekat dengan perkiraan atau dengan kata lain untuk mencari selisih seminimal mungkin antara suatu kejadian dengan apa yang kita perkirakan [14].

Prediksi harga mobil bekas saat ini menjadi trend yang menarik dan popular, karena harga mobil biasanya bergantung pada banyak fitur dan faktor-faktor khusus [4]. Fitur-fitur yang biasanya mempengaruhi harga mobil yaitu usia mobil, model, asal mobil, jarak tempuhnya, tenaga kudanya, jenis bahan bakar, akselerasi, gaya interior, jumlah silinder (diukur dalam cc atau sentimeter kubik), sistem pengereman,

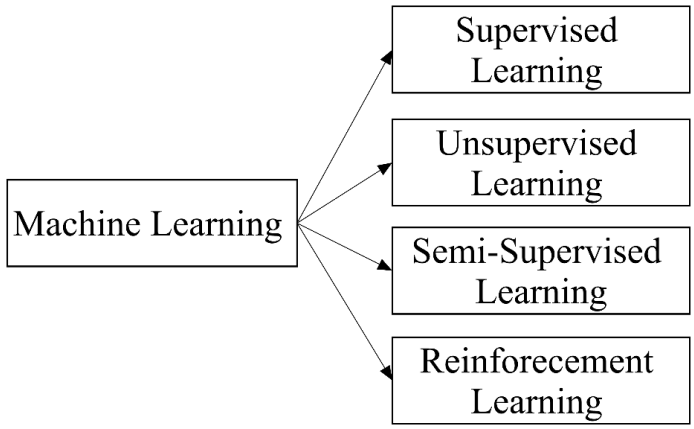
ukurannya, indeks keselamatan, jumlah pintu, berat mobil, warna cat, penghargaan bergengsi yang dimenangkan oleh pembuat mobil, ulasan pelanggan, keadaan fisiknya, jenis transmisi, milik pribadi atau perusahaan, pendingin, roda kosmik , mekanisme kemudi, navigator GPS, kerusakan diperbaiki atau tidak [5]. Oleh sebab itu, harga mobil dipengaruhi oleh banyak fitur dan faktor, namun belum tentu semua fitur tersebut tersedia pada data yang ada, sehingga pemilik *showroom* mobil bekas perlu membuat keputusan untuk menetapkan harga mobil bekas berdasarkan fitur-fitur dan faktor yang ada.

2.1.2 Algoritma *Machine Learning*

Machine learning merupakan bidang ilmu dan juga seni terkait komputer dengan menggunakan teknik statistik untuk melakukan inferensi terhadap kumpulan data yang diberikan, *machine learning* secara umum dijelaskan oleh Arthur Samuel pada tahun 1959 yaitu bidang ilmu yang mampu memberikan pembelajaran kepada komputer tanpa diprogram secara eksplisit [15], [16].

Seorang matematikawan, ahli astronomi serta ahli geografi yaitu Al-Khawarizmi menjelaskan bahwa algoritma merupakan langkah-langkah untuk melakukan perhitungan, memproses data dan penalaran secara otomatis, sedangkan Profesor Harold Stuart Stone mengungkapkan bahwa algoritma merupakan instruksi-instruksi dalam bahasa yang dimengerti oleh komputer untuk proses yang cepat, efisien, baik yang menentukan gerakan komputer [17].

Saat ini, sudah banyak algoritma dari *machine learning*. Namun secara garis besar algoritma *machine learning* dibagi ke dalam 4 kategori yaitu *Supervised Learning, Unsupervised Learning, Semi Supervised Learning, Reinforcement Learning.* [17] seperti yang bisa dilihat di Gambar 2.1.



Gambar 2.1. Tipe Algoritma *Machine Learning*

Sumber : [17]

Machine learning terbagi menjadi 4 bagian, yaitu [17] [15]:

a. *Supervised Learning*

Merupakan pembelajaran yang diawasi/terarah, pembelajaran yang khas dari *supervised learning* yaitu klasifikasi dan prediksi, pada pembelajaran ini data latih yang dimasukkan ke algoritma harus menggunakan label. Pada pembelajaran ini, algoritma klasifikasi dapat digunakan untuk regresi, dan algoritma regresi dapat digunakan untuk klasifikasi, karena mampu menghasilkan nilai yang sesuai dengan probabilitas milik kelas tertentu. Dalam pembelajaran ini diibaratkan seperti seorang guru yang mengajar muridnya, murid dianggap sebagai algoritma dan guru dianggap sebagai data *training*. Proses *training* akan berhenti jika algoritma sudah mencapai kinerja yang optimal (*an acceptable level of performance*).

b. *Unsupervised Learning*

Merupakan pembelajaran tanpa diawasi sehingga data latih yang dimasukkan ke dalam algoritma tidak berlabel. Tugas umum pada pembelajaran *Unsupervised Learning* seperti asosiasi dimana tujuannya adalah menggali data dalam jumlah besar dan menemukan hubungan yang menarik antar atribut, kemudian deteksi anomali dengan cara sistem dilatih dengan contoh normal, dan ketika melihat data baru dapat mengetahui apakah itu terlihat normal atau kemungkinan anomali, selain itu, pembelajaran ini mampu melakukan pengurangan dimensi untuk menyederhanakan data tanpa kehilangan banyak informasi. Salah satu cara

untuk melakukannya adalah dengan menggabungkan beberapa atribut yang berkorelasi. Pada pembelajaran ini dianalogikan seperti orang yang sedang memeriksa lembar jawaban tapi tidak ada kunci jawaban, sehingga cara penilaian jawaban sangat bergantung dengan kemampuan orang tersebut dalam memahami jawaban soal.

c. *Semi Supervised Learning*

Merupakan kombinasi dari pembelajaran *Supervised Learning* dan *Unsupervised Learning* dimana dalam pembelajaran ini hanya menggunakan data berlabel yang sedikit kemudian proses *training* dilanjutkan dengan data tanpa label sehingga algoritma akan belajar mencari pola yang sudah ada dan mencoba mencari pola baru yang belum diketahui.

d. *Reinforcement Learning*

Merupakan pembelajaran yang mampu belajar dari kesalahan dan kemudian harus belajar sendiri strategi untuk mendapatkan kebijakan yang terbaik, sebuah kebijakan mempengaruhi tindakan apa yang harus dilakukan untuk mendapatkan hasil maksimal. *Reinforcement Learning* dipengaruhi oleh umpan balik dari lingkungan dan melakukan proses pembelajaran secara berulang dan *adaptive* layaknya mendekati cara manusia belajar.

2.1.3 Algoritma *XGBoost*

Algoritma *XGBoost* merupakan sistem pembelajaran yang dapat diskalakan untuk meningkatkan sebuah pohon keputusan atau *decision tree*, dan sistem mampu bekerja lebih cepat dibandingkan dengan mesin pembelajaran lain serta dengan pembelajaran pohon baru mampu mengatasi *sparse* data dan pembobotan data [18]. Algoritma *XGBoost* merupakan implementasi yang efisien karena dapat diskalakan , selain itu *XGBoost* mampu melakukan komputasi paralel *multithreading* secara otomatis dan tingkat akurasi prediksi yang tinggi. *Algoritma XGBoost* secara efektif dapat menangani nilai yang hilang, cocok dalam stabilitas transien prediksi untuk menemukan hubungan antara fitur dan stabilitas sementara [19].

XGBoost merupakan salah satu algoritma dengan kecepatan komputasi tinggi serta memiliki dukungan komputasi terdistribusi untuk mengevaluasi model yang kompleks dengan kumpulan data yang besar dan bervariasi di formulasikan dengan metode *ensemble* memberikan model kesalahan yang diantisipasi dari beberapa pohon

keputusan [20]. Metode *ensemble* pada *XGBoost* didasarkan pada *gradient boosting tree* yang telah menjadi alat kompetitif di antara model *artificial intelligence* karena mudah dalam memparalelkan fitur serta akurasi prediksi yang tinggi[18].

Berikut ini merupakan beberapa keuntungan dari *XGBoost*, sebagai berikut [21] :

1. *XGBoost* dikenal sebagai teknik *regularized boosting* dan mampu mengurangi *overfitting*
2. *XGBoost* mengimplementasikan pemrosesan paralel dan jauh lebih cepat dibandingkan dengan *GBM*
3. *XGBoost* memungkinkan pengguna untuk menentukan *custom optimization objectives* dan evaluasi kriteria
4. *XGBoost* mampu menangani data missing value dengan cara memberikan nilai yang berbeda dari pengamatan lain, karena *XGBoost* mempelajari dan mencoba hal baru ketika menemukan nilai yang hilang untuk di masa mendatang
5. *XGBoost* membuat *split* hingga *max depth* yang ditentukan dan membuang pohon kebelakang dan menghapus *split* yang tidak berpengaruh positif
6. *XGBoost* memungkinkan pengguna untuk menjalankan validasi silang(*cross validation*) pada setiap iterasi dari proses *boosting*, dengan demikian mudah untuk mendapatkan jumlah iterasi *boosting* yang optimal dalam satu kali proses.

2.1.4 *Data Preprocessing*

Beberapa *machine learning* tidak dapat bekerja pada data yang kotor seperti data yang hilang dan *outlier* dapat mempengaruhi hasil kinerja model yang dibangun. Teknik pembersihan data mampu mengatasi masalah tersebut dan membuat data siap untuk dijadikan data input dari model yang dibangun. Teknik pembersihan data dapat dilakukan dengan *Missing Values Handling* dan *outlier Detection* [22].

Preprocessing merupakan teknik yang penting dalam penemuan pengetahuan yang didominasi kumpulan data yang besar, Teknik ini penting dilakukan karena dapat mengurangi kompleksitas data sehingga dapat lebih mudah diproses oleh solusi penambangan data atau sering disebut *data mining* [23].

a. Data Cleaning

Tahap data *cleaning* merupakan proses pembersihan kumpulan data yang kotor seperti pengamatan data, penghapusan data yang tidak relevan, menghilangkan data yang duplikat, penandaan data (*marking*), ataupun penambahan data hilang [16].

b. Outlier Detection

Outlier Detection merupakan proses menemukan data yang memiliki perilaku sangat unik dan terlihat sangat berbeda jauh dari observasi lainnya. Deteksi *outlier* mencoba menangkap kasus-kasus luar biasa yang memperlihatkan penyimpangan secara signifikan dari pola mayoritas. Kebanyakan *outlier* disebabkan oleh kesalahan pengukuran atau pencatatan yang salah, sehingga mengabaikan kasus yang jarang muncul [24][22].

Outlier Detection merupakan teknik untuk menemukan data atau kejadian atau item yang berbeda dan mencurigakan dari kumpulan dataset yang ada dan biasanya diluar kisaran normal untuk seluruh data [25]. Oleh karena itu, banyak algoritma telah diusulkan dan dikembangkan untuk mendeteksi *outlier*. Deteksi *outlier* atau pembersihan data merupakan langkah penting dalam pemrosesan data karena, jika titik data *outlier* digunakan selama penambangan data, kemungkinan menghasilkan keluaran yang tidak akurat [23].

2.1.5 *Data Transformation*

Pada tahap ini merupakan proses merubah struktur data menjadi format yang lain untuk mesin komputer lebih mudah memahami data dan dapat diproses [16].

A. *RobustScaler*

Normalisasi data digunakan untuk menstandarkan dari setiap fitur sehingga nilai dari setiap fitur akan berada pada skala yang sama [26]. Dengan merubah distribusi nilai asli menjadi satu set nilai baru dapat membantu daya prediksi yang lebih baik dari model [24].

Robustscaler merupakan pendekatan untuk penskalaan data yang melibatkan penghitungan median dan persentil ke 25 dan persentil ke 75 dari setiap fitur, yang mana nilai dari masing-masing fitur dikurangi nilai mediannya dan dibagi dengan

rentang *interkuartil* (*IQR*) yaitu selisih antara persentil ke 75 dan persentil ke 25. Pendekatan ini mampu menangani data outlier [27].

$$Value = (value - median) / (p75 - p25)$$

B. *Transformation Log*

Transformasi log biasanya digunakan untuk menangani data *time series* yang miring, pendekatan ini mampu mengurangi variabilitas data dan membuat data lebih sesuai dengan distribusi normal [28]. Setiap variabel x diganti dengan log(x) untuk membantu mengompres sumbu y ketika memplot histogram, selain itu dapat menghilangkan penekanan pada *outlier* dan memungkinkan akan memperoleh distribusi berbentuk lonceng [29].

2.1.6 *Feature Selection*

Feature Selection merupakan sebuah proses yang biasa digunakan pada *Machine Learning* yang mana fitur-fitur yang dimiliki oleh sekumpulan data dipakai untuk pembelajaran algoritma. *Feature selection* merupakan salah satu faktor yang penting karena dapat mempengaruhi performa dari sebuah model yang dibangun, dengan semakin banyaknya fitur maka dimensi data akan semakin besar pula sehingga akan mengurangi tingkat performa model [30].

Dalam membangun sebuah model, sering dijumpai terdapat banyak fitur dalam data, fitur-fitur tersebut tidak semua bisa memberikan pengaruh baik terhadap performa model. Fitur yang kurang berpengaruh harus dipilih dan dibuang agar yang digunakan adalah fitur-fitur yang berpengaruh terhadap model. Proses pemilihan fitur disebut juga dengan *feature selection* [11].

Seleksi fitur dilakukan untuk *decreasing the learning cost* atau menurunkan biaya pembelajaran, *increasing the learning performance* atau meningkatkan kinerja pembelajaran dan *reducing irrelevant dimensions* atau mengurangi dimensi yang berlebihan [30].

2.1.7 *Feature Importance*

Feature Importance merupakan proses perhitungan nilai-nilai penting dari setiap fitur yang dimiliki dataset sehingga memungkinkan fitur diberi pemeringkatan untuk

dibandingkan satu sama lain dengan fitur lain, semakin tinggi nilai fitur maka semakin penting pula fitur tersebut [26].

2.1.8 *Hyper-parameter tuning*

Hyper-parameter Tunning merupakan metode untuk pengoptimalan kinerja algoritma pembelajaran mesin yang terdapat beberapa jenis *hyperparameter* yang diatur secara manual untuk meningkatkan kinerja dari sebuah algoritma [31]. Tujuan dari penggunaan *hyperparameter tuning* untuk mencari nilai-nilai terbaik dari parameter yang disetting agar meminimalkan fungsi tujuan yang didefinisikan dalam persamaan, dan telah terbukti lebih efisien diterapkan untuk data yang memiliki dimensi tinggi [32].

2.1.9 Parameters

Berikut ini merupakan beberapa parameter yang digunakan untuk algoritma *XGBoost* pada *hyperparameter tuning* :

1. *Learning rate*

   Learning rate merupakan parameter *training* yang digunakan untuk mengatur seberapa pengaruh keterjalan terhadap proses *training* dan secara perlahan mampu memberikan nilai yang optimal dari perubahan bobot sehingga akan menghasilkan nilai kesalahan yang lebih kecil. Nilai dari sebuah *learning rate* menyatakan lama waktu untuk proses belajar dari sebuah model untuk setiap iterasinya, nilai *rate* berkisar antara 0 sampai dengan 1. Semakin besar nilai *learning rate* yang diberikan, maka proses pembelajaran akan semakin cepat. Namun, jika nilai *learning rate* terlalu besar terdapat kelemahan yaitu dapat menyebabkan jaringan tidak stabil dan nilai eror akan berulang diantara nilai tertentu [7] [32]. Oleh karena itu, pemberian nilai sebuah *learning rate* harus sebaik mungkin agar mendapatkan proses pembelajaran data *training* yang optimal dan cepat.
2. *Max_depth*

   Max depth merupakan parameter yang digunakan untuk mengendalikan model kompleksitas guna mencegah terjadinya *overfitting* [34]. *Max depth* merupakan jumlah maksimum *node* yang diperbolehkan dari *root* ke daun terjauh dari sebuah pohon. Pohon yang lebih dalam dapat memodelkan

hubungan yang lebih kompleks dengan menambahkan lebih banyak *node*, tetapi saat kita masuk lebih dalam, pemisahan menjadi kurang relevan dan terkadang hanya karena *noise*, yang menyebabkan model menjadi *overfit* sehingga harus disetel menggunakan CV [35],[21].

3. *n_estimators*

   n_estimators merupakan jumlah pohon yang didorong oleh *gradien*. Setara dengan jumlah putaran *boosting*. Parameter ini menentukan berapa kali harus melalui siklus pemodelan yang dijelaskan[36]

4. *Subsample*

   Subsample menunjukkan fraksi pengamatan menjadi sampel acak untuk setiap pohon, Ketika menyetelnya ke nilai 0,5 berarti algoritma akan mengambil sampel secara acak setengah dari data pelatihan sebelum membangun pohon. Hal Ini mampu mencegah terjadinya *overfitting*. *Subsampling* akan terjadi sekali dalam setiap iterasi *boosting* dan rentang nilai biasanya antara 0 sampai dengan 1 [36].

5. *Min_child_weight*

   Mendefinisikan jumlah minimum bobot dari semua pengamatan yang diperlukan pada seorang anak(*child*). Digunakan untuk mengontrol *overfitting*. Nilai yang lebih tinggi mencegah model dari hubungan pembelajaran yang mungkin sangat spesifik untuk sampel tertentu yang dipilih untuk pohon. Nilai yang terlalu tinggi dapat menyebabkan *underfitting* sehingga harus disetel menggunakan *CV* [21]. Jika langkah partisi pohon menghasilkan simpul daun dengan jumlah bobot *instance* kurang dari *min_child_weight*, maka proses pembangunan akan menghentikan partisi lebih lanjut [36].

6. *Seed*

   Seed merupakan nilai acak dari benih yang digunakan untuk menghasilkan hasil yang dapat direproduksi serta memastikan penggunaan lipatan yang sama untuk setiap langkah sehingga dapat membandingkan skor dengan parameter yang berbeda dengan benar serta dapat juga digunakan untuk parameter *tuning* [35] [21].

7. *n_job*

Merupakan jumlah utas paralel yang digunakan untuk menjalankan *XGBoost*. Saat digunakan dengan algoritme *Scikit-learn* lainnya seperti *grid search*, akan mampu memilih algoritma mana yang akan diparalelkan dan diseimbangkan utasnya. Membuat pertikaian utas akan memperlambat kedua algoritma secara signifikan [36].

8. *Colsample_bytree*

   Colsample Bytree merupakan rasio subsampel fitur saat membangun setiap pohon dan termasuk dalam kelompok parameter *subsampling* yang memiliki nilai range 0 sampai dengan 1. Ketika menggunakan nilai 1 secara *default* akan menggunakan semua fitur [36].

9. *Objective*

   Objective merupakan parameter yang digunakan untuk menentukan tujuan pengoptimalan, metrik yang akan dihitung pada setiap langkah. Mereka digunakan untuk menentukan tugas pembelajaran dan tujuan pembelajaran yang sesuai. Berikut beberapa contoh jenis pilihan objective [21][36]:

   a. *reg:linear*
   b. *reg:squarederror*
   c. *reg:squaredlogerror*
   d. *reg:logistic*
   e. *binary:logistic*
   f. *multi:softmax*

2.1.10 *Cross Validation*

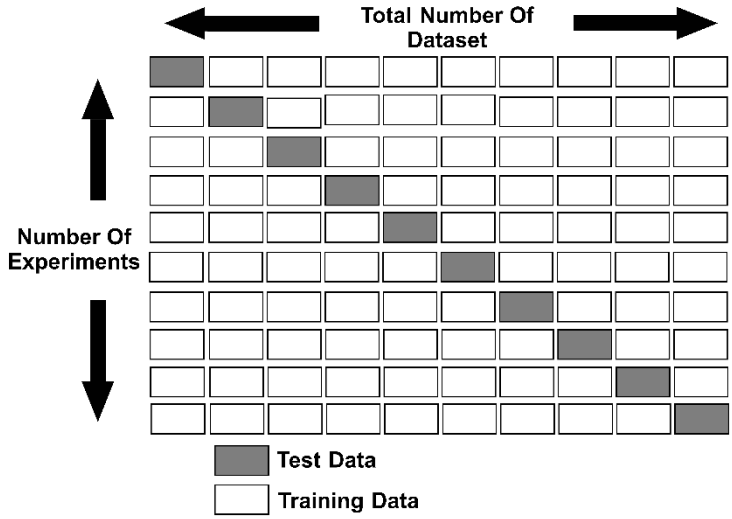
Cross Validation merupakan strategi pemisahan silang untuk pengambilan sampel ulang data yang tersedia untuk mengevaluasi model pembelajaran mesin yang mana banyaknya grup data ditentukan oleh parameter K. penggunaan *cross validation* dapat mengurangi bias data atau mengoptimalkan hasil dibandingkan dengan *train-test split* sederhana [37].

Cross validation merupakan teknik validasi model secara silang untuk menilai apakah hasil statistik dari analisis yang dilakukan sudah menggeneralisasi dari sekumpulan data independent dan mampu mengetahui seberapa akurat dari hasil prediksi yang dilakukan. *Cross validation* digunakan untuk mencari nilai *error rate*

dengan cara membagi sekumpulan data menjadi data latih (*training*) dan validasi yang kemudian akan dievaluasi kinerja algoritmanya [16].

Untuk mengetahui nilai *error* dari model yang dibuat *cross validation*, semakin kecil nilai *error rate* yang dihasilkan maka model tersebut dikatakan semakin baik begitu sebaliknya.

Beberapa kategori *cross validation* yaitu *Hold out method, K-Fold CV, Leave one out CV* (LOO CV), dan *Bootstrap method* [16].



Gambar 2.1 *Fold Cross Validation*

Sumber : [38]

a. *Hold out method*

Merupakan teknik dengan membagi data ke dalam *data train* dan *data test*, yang dilakukan secara acak dan pengujian *error rate* dilakukan dengan mencari nilai *RMSE (Root Mean Square Error*). Jika nilai *RMSE* antara data *training* dan data *testing* hampir sama, model dapat dikatakan bagus dan begitu sebaliknya. Meskipun sederhana tetapi metode *Hold out method K-Fold CV* ini kurang direkomendasikan karena peluang terjadi bias cukup besar dan biasanya cocok untuk data yang mempunyai variasi sampel tidak jauh berbeda.

b. *K-Fold CV*

Pada metode ini, data dibagi ke dalam set K partisi dan jumlah sampel partisi yang sama. Pemilihan sampel untuk setiap partisi dilakukan secara acak, dan nilai K

yang digunakan biasanya berkisar antara 5 sampai dengan 10. *K-Fold* mampu mengurangi waktu komputasi dan mengurangi peluang terjadinya bias.

$$k = \frac{N}{K}$$

Dimana :

K = Jumlah partisi

N = Jumlah Sampel (data)

K = Jumlah sampel masing-masing partisi

c. *Leave one out CV (LOO CV)*

Merupakan metode yang hampir mirip dengan *K-Fold CV* tetapi lebih ke dalam kasus yang lebih khusus, karena dalam metode ini K = N, dimana N adalah jumlah data yang ada sehingga jumlah partisi hanya akan memiliki sampel saja dan akan memakan waktu komputasi yang lama jika jumlah datanya semakin banyak. Namun metode ini cocok untuk data validasi karena data diuji satu persatu.

d. *Bootstrap method*

Dengan metode ini akan dilakukan pembentukan data set baru dari data yang ada dan biasa disebut dengan *bootstrap dataset* yang mana setiap *bootstrap* akan memiliki data dengan jumlah yang sama dengan data asli dan kemudian model yang dibuat akan dicocokkan dengan *bootstrap dataset*.

2.1.11 Metode *Grid Search*

Grid Search merupakan metode untuk penyetelan dan mencari parameter terbaik dari sebuah model dengan mengeksplorasi nilai-nilai yang diberikan, yang mana metode *Grid Search* akan membangun dan mengevaluasi model untuk setiap kombinasi parameter [37]. Metode *Grid Search* mampu menentukan parameter terbaik. dalam hal ini, algoritma yang menjadikan titik-titik *grid* berjarak sama dan kemudian menghitung kesalahan untuk setiap titik parameternya,titik yang paling optimal yaitu dengan nilai kesalahan yang terkecil [39].

2.1.12 *Feature Importance*

Feature Importance merupakan cara untuk mencari atau menunjukkan besarnya fungsi atau pengaruh dari setiap fitur data yang digunakan untuk membangun model, *Feature Importance* mampu menampilkan peringkat dari setiap fitur kemudian dapat dibandingkan dengan fitur lainnya [34].

2.1.13 Performa Peramalan

Setelah mendapatkan model terbaik, model harus dicek apakah sudah baik atau belum. Proses untuk mengetahui performa model dapat digunakan evaluasi kinerja, setelah model dilatih dengan data *training* kemudian model dilatih dengan data baru. Proses tersebut disebut proses *scoring* yaitu memanfaatkan model dengan data baru [11].

Evaluasi kinerja dalam pemodelan prediksi digunakan untuk membandingkan model yang dilatih dengan data actual/asli dari kumpulan data pengujian untuk mengukur seberapa jauh perkiraan menyimpang (nilai eror) dari pengamatan untuk menilai kualitas dan memilih metode peramalan terbaik [40].

1. *Mean Squared Error*

Mean Square Error (*MSE*) merupakan indikator yang digunakan untuk mengukur hasil *running* dari sebuah model. Fungsinya untuk mengevaluasi performa model selama proses pelatihan [41]. *MSE* ini merupakan metrik paling sederhana untuk evaluasi regresi, *MSE* mengukur kesalahan kuadrat rata-rata dari prediksi, *MSE* menghitung selisih kuadrat antara hasil prediksi dan target kemudian merata-rata [42].

$$MSE = \sqrt{\frac{\sum_{i=1}^{n}(xi - E(i)^2}{n}} \quad (1)$$

Dimana :

MSE = Mean Square Error

n = Jumlah data

xi = parameter *input*

E(i) = Target *output*

Semakin tinggi nilai *MSE* yang dihasilkan, maka semakin buruk model yang dibuat. *MSE* tidak pernah bernilai negatif karena perhitungannya dengan

mengkuadratkan kesalahan prediksi dan akan menjadi nol yang berarti model tersebut sempurna [42].

2. *Mean Absolute Error (MAE)*

Mean Absolute Error (MAE) merupakan salah satu metode evaluasi yang digunakan untuk mengukur tingkat keakuratan model prediksi yang mana nilai *MAE* tersebut menunjukkan rata–rata kesalahan absolut antara hasil prediksi dengan nilai asli [42]. Berikut ini merupakan rumus *Mean Absolute Error (MAE)* [43].

$$MAE = \frac{1}{n}\sum_{i=1}^{n} |f_i - y_i| \quad (2)$$

n = Jumlah data

fi = Nilai hasil prediksi

yi = Nilai sebenarnya

Rumus tersebut digunakan untuk mencari nilai *MAE* dengan menghitung nilai rata-rata dengan memberikan bobot yang sama untuk seluruh data secara intuitif. Dengan pemberian bobot yang sama pada semua data dapat memberikan nilai yang lebih tepat [43]

3. *R2 Score*

Pengujian model dengan data test digunakan untuk mengukur performa model dalam memprediksi, pengujian dapat dilakukan dengan fungsi *score* atau sering ditulis dengan *R2* dan akan menghasilkan nilai *R-Squared* dimana nilai yang mendekati nilai 1 artinya model tersebut semakin baik dan mendekati 0 yang berarti model semakin buruk. Nilai *R-Squared* menjadi patokan seberapa dekat sebuah data dengan garis lurus.[11].

Dimana Rumus yang digunakan yaitu :

$$\textbf{\textit{R-Squared}} = \textbf{1-(RSS/TSS)} \quad (3)$$

Dimana *Residual Sum of Squared (RSS)* merupakan hasil dari menjumlahkan kuadrat dari semua *residual* dan Total *Sum of Squared (TSS*) merupakan hasil dari menjumlahkan kuadrat dari selisih angka prediksi dengan nilai rata-rata.

Nilai *R2* dipengaruhi oleh nilai Y prediksi. Nilai *R2* akan semakin membaik jika nilainya terus mendekati nilai 1 dan jika kita menambah fitur. Semakin banyak jumlah

fitur yang menentukan nilai Y prediksi, maka nilai *RSS* akan semakin besar dan akibatnya nilai *R2* juga akan meningkat [44].

2.1.14 *Tools Python*

Pada penelitian ini, digunakan *Google Colab* sebagai media untuk mengolah data dan membangun model *machine learning*, *google colab* merupakan layanan penyimpanan gratis yang diberikan oleh *Google* dengan berbasis *Jupyter Notebook*. Bahasa pemrograman yang digunakan yaitu *python*.

2.2 Tinjauan Studi

Beberapa penelitian yang telah dilakukan terkait dengan prediksi harga mobil bekas adalah sebagai berikut :

1. Penelitian yang dilakukan oleh [8] mengenai prediksi harga mobil bekas dengan model *regresi* dengan judul *Prediction of prices for used car by using regression models* untuk memprediksi harga mobil bekas dengan algoritma *multiple linear regression, random forest regression*, and *gradient boosted regression trees*. Setiap model dievaluasi dengan data pengujian yang sama menghasilkan kinerja terbaiknya berada pada algoritma *Gradient Boosted Regression tree* dengan *Mean Squared Error(MSE*)=0.28, Diikuti dengan *Random Fores*t sebesar *MSE*=0.35,dan *Multiple Linear Regression* sebesar *MSE*=0.55. Dari penelitian tersebut terdapat *novelty* yaitu terjadi penurunan tingkat *error* ketika menggunakan algoritma *Gradient Boosted Regression tree* . *Gradient Boosted Regression tree* direkomendasikan untuk membangun model evaluasi pada prediksi harga mobil bekas.
2. Penelitian yang dilakukan oleh [5], Membahas tentang prediksi harga mobil bekas menggunakan Algoritma Pembelajaran Mesin yaitu *Regresi Linier*. Data nol dan data yang hilang dihapus serta penggunaan *One Hot Encoding* untuk mengolah variabel yang bertipe kategori. Hasil menunjukkan bahwa terdapat korelasi positif antara Harga Jual dan Harga Sekarang sedangkan korelasi negatif antara Harga Jual dengan *Kms Driven*, Tahun dan Pemilik (Jumlah Pemilik Sebelumnya). Selain itu, hasil menunjukkan bahwa harga jual mobil lebih tinggi jika dijual oleh *dealer* dibandingkan dengan individu. Demikian pula harga jual lebih tinggi untuk mobil yang bertransmisi otomatis. Terlihat

juga bahwa harga jual mobil dengan Jenis Bahan Bakar Diesel lebih tinggi dibandingkan dengan jenis Bahan Bakar Minyak dan CNG. R2 skor pada *Regresi Linier* adalah 0,86 yang berarti baik dan prediksi cukup mendekati harga jual semula

3. Penelitian yang dilakukan oleh [4]. Penelitian mengenai prediksi harga mobil dengan teknik *machine learning* dengan judul *Car Price Prediction using Machine Learning Techniques*, Penelitian menggunakan tiga teknik *Machine Learning (ANN, SVM* dan *Random Forest*) model ensemble diterapkan untuk membangun model prediksi harga mobil bekas di Bosnia dan Herzegovina. Model prediksi akhir diintegrasikan ke dalam aplikasi Java. Selanjutnya model dievaluasi menggunakan data uji dan diperoleh akurasi sebesar 87,38%. Namun, kelemahan dari sistem yang diusulkan adalah menghabiskan lebih banyak sumber daya komputasi daripada algoritma pembelajaran mesin tunggal.
4. Penelitian yang dilakukan oleh [45], yang membahas tentang prediksi harga mobil dengan menggunakan pembelajaran mesin yaitu *multiple linear regression*, dengan mengusulkan sebuah sistem yang mana variabel harga dijadikan variabel prediksi dan variabel harga berasal dari faktor-faktor seperti model kendaraan, merek, kota, versi, warna, jarak tempuh, *velg*, dan *power steering*. Model dibentuk dengan menetapkan 23 dari 78 variabel yang mempengaruhi harga. Tingkat Determinasi (*R2 Score*) dari 23 variabel ini sebesar 89,1%. Dengan menggunakan model *multiple linear regression*, diperoleh kecocokan antara nilai asli dan nilai prediksi. Menurut hasil penelitian ini, terlihat bahwa penggunaan teknik pembelajaran mesin sudah tepat untuk memprediksi harga mobil bekas.
5. Penelitian yang dilakukan oleh [46] membahas tentang harga mobil bekas dengan mengembangkan model statistik dari Algoritma pembelajaran mesin seperti *Lasso Regression, Multiple Regression* dan *Regression trees* berdasarkan data konsumen sebelumnya dan serangkaian fitur. Hasil menunjukkan, tingkat kesalahan prediksi dari semua model jauh di bawah kesalahan 5%. Namun,dengan menggunakan *One-way Analysis Of Variance (ANOVA)* dapat memverifikasi apakah tingkat kesalahan ketiga model berbeda,

dan kesalahan rata-rata pada model *Regression trees* ditemukan lebih besar daripada tingkat kesalahan rata-rata model *Multiple Regression* dan *Lasso Regression*. Untuk mendapatkan gambaran yang lebih jelas, dilakukan juga *post-hoc test* untuk menemukan kelompok-kelompok yang memiliki mean yang berbeda nyata dengan melakukan uji Tukey (*Tukey's Honest Significant Difference Test*) untuk mengetahui kelompok mana yang sebenarnya berbeda satu sama lain. Tes membandingkan semua kemungkinan pasangan rata-rata dan memeriksa perbedaan yang signifikan secara statistik di antara ketiga model. model *Lasso Regression* dan model *Multiple Regression* tidak berbeda secara signifikan, tetapi tingkat kesalahan rata-rata *Regression trees* lebih tinggi dan berbeda secara signifikan dari kedua model. Nilai *Mean Error Rates* dari model *lasso* sebesar 3.5%, model *multiple* sebesar 3.4%dan model *Regression trees* sebesar 3.7%. Hasil dari prediksi mobil bekas ini mampu memberikan pengetahuan kepada pelanggan dan penjual untuk lebih mengetahui tentang tren dan pola yang menentukan nilai mobil bekas di pasar

Tabel 2.1 Penelitian Terkait

No	Penulis	Judul	Metode	Hasil
1	Nitis Monburinon, dkk(2018)	Prediction of Prices for Used Car by Using Regression Models	*multiple linear regression, random forest regression*, and *gradient boosted regression trees*	kinerja terbaiknya berada pada algoritma *Gradient Boosted Regression tree* dengan *Mean Squared Error(MSE*)=0.28, Diikuti dengan *Random Fores*t sebesar *MSE*=0.35,dan *Multiple Linear Regression* sebesar *MSE*=0.55
2	Ashutosh Datt Sharma, Vibhor Sharma(2020)	Used Car Price Prediction Using Linear Regression Model	*Regresi Linier*	R2 skor pada *Regresi Linier* adalah 0,86 yang berarti baik dan prediksi cukup mendekati harga jual semula
3	Enis Gegic, dkk(2019)	*Car Price Prediction using Machine Learning Techniques*	*ANN, SVM* dan *Random Forest*	Diperoleh akurasi sebesar 87,38%. Namun, kelemahan dari sistem yang diusulkan adalah menghabiskan lebih banyak sumber daya komputasi daripada algoritma pembelajaran mesin tunggal
4	Ozer Celik dan U. Omer Osmanoglu (2019)	Prediction of The Prices of Second Hand Cars	*Multiple linear regression*	Model dibentuk dengan menetapkan 23 dari 78 variabel yang mempengaruhi harga. Tingkat Determinasi (*R2 Score*)

				dari 23 variabel ini sebesar 89,1%
5	Venkatasubbu dan Mukkesh Ganesh (2019)	Used Cars Price Prediction using Supervised Learning Techniques	*Lasso Regression, Multiple Regression* dan *Regression trees*	Nilai *Mean Error Rates* dari model *lasso* sebesar 3.5%, model *multiple* sebesar 3.4%dan model *Regression trees* sebesar 3.7%. Hasil dari prediksi mobil bekas ini mampu memberikan pengetahuan kepada pelanggan dan penjual untuk lebih mengetahui tentang tren dan pola yang menentukan nilai mobil bekas di pasar

2.3 Tinjauan Organisasi/Obyek Penelitian

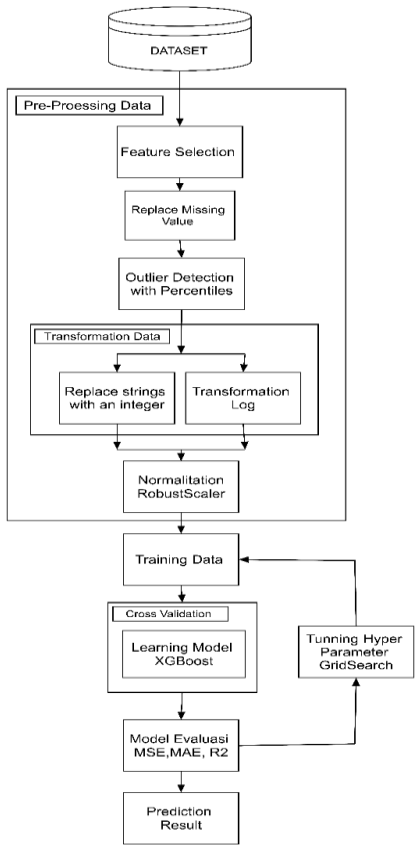
Obyek yang dijadikan bahan untuk penelitian ini menggunakan data iklan penjualan mobil bekas yang diambil dari link *https://data.world/data-society/used-cars-data* [10] dibawah lisensi publik. Data yang diambil merupakan data iklan penjualan mobil bekas dengan jumlah 371.539 record dan 20 attribut diambil dari *Ebay-Kleinanzeigen*, sebuah situs *online* penjualan mobil di Jerman.

BAB III
METODOLOGI PENELITIAN

3.1 Langkah-langkah Penelitian

Penelitian ini menggunakan jenis penelitian kuantitatif yang mana data-data berupa angka, tabulasi dan perhitungan statistik dalam bentuk tabel. Selanjutnya langkah-langkah analisis dapat dilakukan ketika data sudah terkumpul lengkap dan tersaji dalam tabulasi sehingga siap diolah. Penelitian ini dilakukan dengan membuat dan menguji kebenaran dari hipotesis statistik yang melibatkan analisis beberapa variabel dengan tes tertentu. Tujuan dari penelitian ini adalah mendapatkan model algoritma *XGBoost* yang memberikan hasil terbaik dalam memprediksi harga penjualan mobil bekas serta menganalisa hubungan antar atribut terhadap hasil prediksi harga penjualan mobil bekas.

Penelitian ini dilakukan dengan beberapa langkah atau tahapan penelitian. Berikut ini merupakan langkah-langkah penelitian dan digambarkan ke dalam kerangka pemikiran sebagai berikut :



Gambar 3.1 Metodologi Penelitian

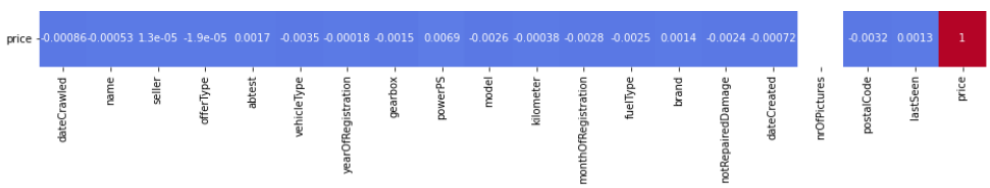
Berdasarkan gambar diatas, langkah-langkah pemikiran dari penelitian yang dilakukan sebagai berikut :

1. *Preprocessing* Data

   Pada tahap ini dilakukan beberapa *Preprocessing* data yaitu dengan menggunakan *Feature Selection, Replace Missing Value, Outlier Detection, Data Transformation,* kemudian *Log transformation* digunakan untuk mentransformasi data agar mengurangi *skewness* yang terjadi pada data. Setelah itu dilakukan proses normalisasi dengan metode *RobustScaler,* dengan menggunakan nilai *median* dan *quartiles,* tujuannya agar tidak rentan terhadap data *outlier*. Tahap *preprocessing* ini digunakan untuk mempermudah dalam proses eksperimen terhadap data latih untuk menghasilkan model yang lebih baik. Selain itu, tahap ini digunakan untuk mendapatkan prediksi dengan nilai eror yang lebih kecil. Berikut ini tahap *preprocessing* dalam penelitian ini :

   A. Feature Selection

   Pada tahap ini, penulis memilih fitur dari data untuk menentukan fitur apa saja yang nantinya akan berpengaruh terhadap model prediksi yang dibuat. Tahap ini akan memilih sebagian saja fitur yang ada pada data untuk diproses dengan dilakukan proses transformasi untuk merubah tipe data *string* menjadi tipe data numerik. Karena tidak semua fitur relevan dengan masalah yang dihadapi, bahkan karena fitur tersebut akan mengganggu dan mengurangi performa maka fitur yang tidak terpakai tersebut dihapus untuk meningkatkan performa. *Feature Selection* dalam penelitian ini menggunakan metode *matrix correlation*. *Matrix correlation* berguna untuk menginvestigasi seluruh hubungan antar variabel numerik dalam dataset.



Gambar 3.2 Korelasi antar atribut dengan atribut target

Pada gambar 3.2 merupakan nilai korelasi setiap atribut terhadap atribut target yaitu harga. Terlihat untuk atribut *nrOfPictures* sama sekali tidak memiliki korelasi dengan atribut apapun, sehingga atribut tersebut perlu dihapus karena akan mengganggu performa. Selain itu ada beberapa atribut seperti *dateCrawled, name, seller, offertype, monthOfRegistration, dateCreated*, dan *lastSeen* juga dihapus karena memiliki nilai korelasi yang cukup kecil dan kurang relevan dengan masalah prediksi harga. Terlihat pula bahwa di bagian harga nilainya menunjukkan angka 1 karena atribut tersebut berkorelasi sempurna dengan dirinya sendiri

B. *Replace Missing Value*

Pada tahap ini, dilakukan proses pembersihan data yang mengandung nilai *NaN* atau data hilang, proses pembersihan data dilakukan dengan 2 cara yaitu dengan menghapus *NaN* pada atribut yang memiliki jumlah *NaN* sedikit dan yang kedua adalah dengan mengganti data *NaN* ke dalam *field* baru seperti "*other*", dan "*Unspecified*", karena data tersebut bertipe kategorikal dengan jumlah *NaN* pada atribut tersebut sangat tinggi. Sehingga perlu dipertimbangkan untuk tidak menghapusnya dan mengganti ke dalam *field* baru.

C. *Outlier Detection with Percentiles*

Mengidentifikasi masalah data yang *outlier* merupakan salah satu bagian yang tidak mudah dilakukan dalam pembersihan data, karena tidak semua nilai-nilai yang diidentifikasi merupakan *outlier* dan harus dihilangkan, Oleh karena itu untuk menangani data yang *outlier* perlu digunakan beberapa metode yaitu dengan penerapan statistik untuk mengidentifikasi dan menghilangkan data yang berada di kisaran tidak normal dari kumpulan data.

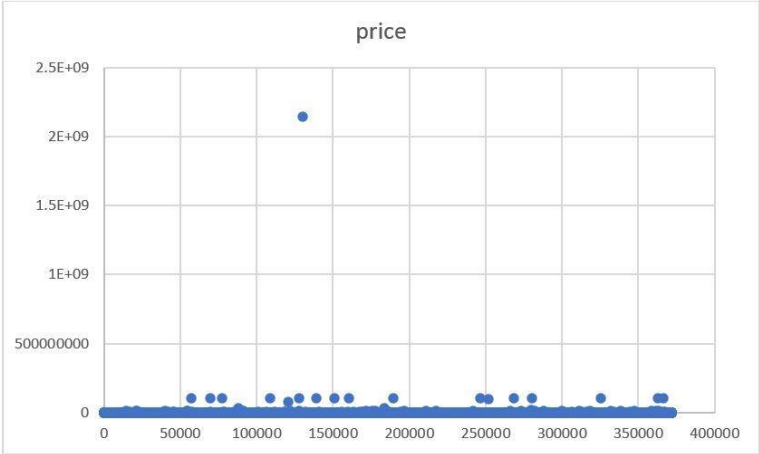
Namun, sebelum proses deteksi *outlier* dilakukan, yang terjadi pada atribut kilometer adalah bertipe kategorikal. Hal tersebut tidak sewajarnya, karena atribut kilometer berisi data angka yang menunjukkan besaran jarak tempuh mobil sehingga perlu dilakukan perubahan tipe data kategorikal menjadi numerik dengan teknik *astype().* Dengan teknik tersebut dapat merubah atribut kilometer menjadi tipe numerik.

Untuk melihat nilai distribusi kemiringannya digunakan *skewness* dan untuk menampilkan penyebaran data *outlier* digunakan *scatter plot*. Dari hasil *skewness* dan *scatter plot* yang ditunjukkan, nampak penyebaran data yang tinggi pada atribut harga, kekuatan kuda dan tahun pembuatan mobil tidak merata dengan baik.

Tabel 3.1 *Skewness* dari Atribut Numerik

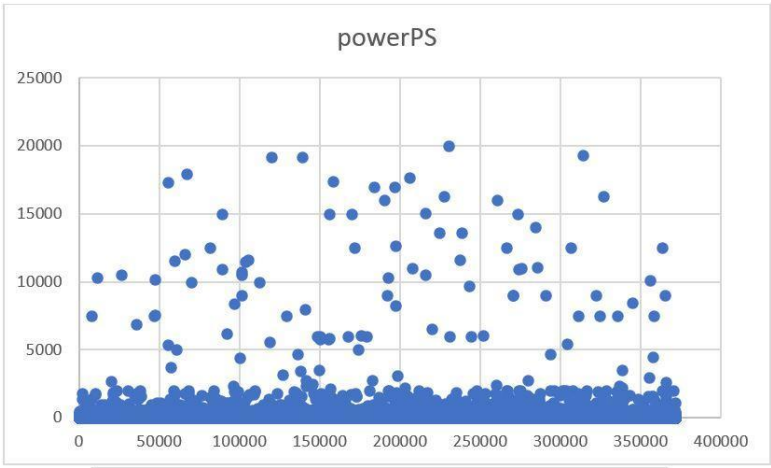
Atribut	***Skew* setelah deteksi *outlier***
price	578,06
yearOfRegistration	72,13
powerPS	58,20
kilometer	-1,55
postalCode	0,06

Tabel 3.1 merupakan hasil dari *skewness* ke lima atribut numerik, terlihat bahwa distribusi kemiringan terjadi pada atribut *price, powerPs*, dan *yearOfRegistration.* Dari ke tiga atribut tersebut kemudian akan dilakukan proses deteksi *outlier* dengan teknik persentil.



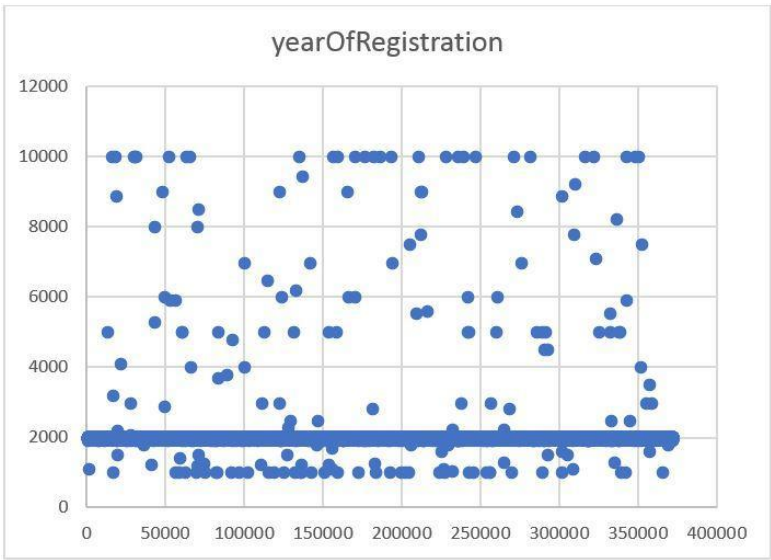
Gambar 3.3 Distribusi Harga dengan *Scatter Plot*

Pada gambar 3.3 merupakan distribusi penyebaran harga mobil bekas, dimana terlihat adanya angka yang sangat tinggi berada jauh diluar penyebaran data normal. Harga tersebut berada di angka lebih dari 2 milyar dan dapat dikatakan sebagai *outlier*. Distribusi harga tersebut dapat mengganggu proses analisa data serta akan berdampak pada pengambilan keputusan dari hasil uji statistik.



Gambar 3.4 Distribusi *powerPs* dengan *Scatter Plot*

Pada gambar 3.4 merupakan distribusi dari *powerPS* atau kekuatan kuda pada mobil, Nampak terjadi distribusi data yang kurang baik, karena dari data sekitar 371.000 rata-rata distribusi berada dibawah angka 5000. Sehingga perlu dilakukan penanganan *outlier* terhadap distribusi *powerPs* tersebut.



Gambar 3.5 Distribusi *yearOfRegistration*

Pada gambar 3.5 merupakan distribusi dari *yearOfRegistration* yang mana terlihat bahwa penyebaran paling banyak terjadi di kisaran tahun 2000. Kemudian nampak terdapat penyebaran data di tahun lebih dari 3000, sedangkan secara logika kita masih berada ditahun 2000 an. Oleh karena itu

perlu dilakukan penanganan terhadap data tahun agar distribusinya lebih mendekati normal.

D. Transformasi Data

a. Transformasi Log

Pada tahap ini, dilakukan proses transformasi log dengan menganalisa tingkat kemiringan kurva lonceng yang terjadi pada fitur. Transformasi log melakukan transformasi data dengan mengganti setiap attribut x dengan *log(x)* sehingga dengan transformasi log ini akan membuat data senormal mungkin. Dengan kata lain, dengan dilakukan proses transformasi log ini akan mengurangi kemiringan data asli, sehingga hasil analisis statistik dari data ini menjadi lebih valid. Berikut tabel *skew* dari hasil deteksi *outlier* dengan teknik persentil pada atribut harga:

Tabel 3.2 *Skew* hasil deteksi *outlier*

Atribut	***Skew* Asli**	***Skew* setelah persentil**	***Skew* Setelah Transformasi log**
price	578.06	1.38	-0,06
yearOfRegistration	72.13	0.23	-
powerPS	58.20	0.62	-

Pada tabel 3.2 merupakan distribusi kemiringan dari 3 atribut yaitu harga, tahun registasi mobil, dan kekuatan kuda mobil. setelah dilakukan proses deteksi *outlier* dengan persentil dan setelah dilakukan proses transformasi log. Pada tahap ini penulis menggunakan *numpy* yang merupakan paket dasar untuk komputasi ilmiah dengan *Python*. Kemudian dengan *numpy* tersebut, penulis menerapkan fungsi *log transformation* yaitu *np.log* untuk mengurangi distribusi kemiringan pada atribut harga sehingga kemiringan akhir pada atribut harga sebesar -0,06 dan untuk atribut tahun serta kekuatan kuda tidak dilakukan proses transformasi *log* karena tingkat kemiringan sudah mendekati normal.

b. *Replace String with Integer*

Pada tahap ini, penulis melakukan proses perubahan tipe data dari kategori menjadi tipe numerik, karena pada beberapa kasus model pembelajaran mesin hanya dapat memproses data dengan tipe numerik. Sehingga proses

ini perlu dilakukan agar model dapat memproses data yang diberikan dan mengekstrak informasi yang berharga untuk prediksi harga mobil bekas.

E. Normalization

Pada tahap ini, penulis melakukan proses normalisasi data dengan *RobustScaler*, yang mana dalam prosesnya menggunakan rentang nilai median dan kuartil agar data tahan terhadap *outliers*. Teknik *scaler* ini akan membuat pusat data ke dalam rentang nilai yang sama, dan bentuk distribusinya tetap sama, sehingga ketika model pembelajaran mesin memproses data akan menemukan nilai dari setiap fiturnya lebih mudah dan cepat.

2. Training Data

Pada tahap ini, dilakukan proses pemisahan data menggunakan *cross validation*. Pada proses pengambilan sampel dilakukan sedemikian rupa agar tidak ada dua set tes yang tumpang tindih. Dalam *k-fold cross-validation*, *learning set* yang tersedia dipartisi menjadi k *subset* yang terpisah dengan ukuran yang kira-kira sama. Di sini, "k" mengacu pada jumlah himpunan bagian yang dihasilkan dengan pengambilan sampel kasus secara acak dari dataset. Model dilatih menggunakan k himpunan bagian, yang bersama-sama mewakili himpunan pelatihan. Kemudian, model tersebut diterapkan pada subset yang tersisa, yang dilambangkan sebagai set validasi, dan kinerjanya diukur. Prosedur ini diulang sampai masing-masing dari k subset telah berfungsi sebagai set validasi [47].

3. Model dengan *Grid Search Cross Validation*

Pada tahap ini, dilakukan uji coba berbagai model terhadap data yang ada dengan memilih model yang memiliki performa paling baik. Untuk memperoleh hasil prediksi yang akurat tidak cukup dengan algoritma *machine learning* saja, tetapi diperlukan beberapa parameter untuk mengatur proses pembelajaran mesin. Pada tahap ini digunakan *hyperparameter tuning* untuk menentukan kombinasi yang tepat dari model dan *hyperparameter* .

Dengan uji coba satu persatu kombinasi yang telah diatur dan ditambah dengan *Grid SearchCV* akan mampu menentukan kombinasi mana yang menghasilkan performa terbaiknya. Pada penelitian ini,nilai *CV* disetting 10 yang mana setiap kombinasi model dan parameter divalidasi sebanyak 10 kali dan data akan dibagi

menjadi 10 bagian sama besar secara acak, yang mana 9 bagian akan digunakan untuk *training* dan 1 bagian untuk *testing*.

4. Model Evaluasi

   Model yang sudah diperoleh berdasarkan data pengujian untuk prediksi harga mobil bekas akan menampilkan nilai kesalahan atau *error*, untuk menampilkan nilai *error* tersebut digunakan metode evaluasi model dengan menggunakan *Mean Squared Error (MSE), Mean Absolute Error (MAE) dan* R^2 *score*.

5. Hasil Prediksi

   Setelah melakukan beberapa uji coba atau eksperimen, didapat hasil dan model terbaik dengan nilai akurasi tertinggi untuk prediksi harga mobil bekas, sehingga pada tahap ini akan menampilkan hasil dari prediksi model yang telah dibangun menggunakan *XGBoost* dengan *Grid Search Cross Validation* terhadap harga mobil bekas.

3.2 Metode Pengumpulan Data

Pengumpulan data dapat dilakukan dengan dua cara yaitu pengumpulan data primer dan pengumpulan data sekunder yang mana data primer dapat diperoleh dengan cara observasi, wawancara, kuesioner dll, yang dikumpulkan langsung dari sumber data. Pengumpulan data primer memerlukan waktu dan biaya yang lebih banyak dari data sekunder. Sedangkan pengumpulan data sekunder dapat diperoleh langsung dari orang lain baik yang sudah dipublikasi maupun yang belum dipublikasi seperti buku, jurnal, dokumentasi, dll yang berhubungan dengan masalah yang akan diteliti. data tersebut sebelumnya digunakan dengan tujuan yang berbeda. Data sekunder relatif lebih cepat dan biaya rendah [48] [49].

Pada penelitian kali ini, teknik pengumpulan data yang dilakukan dengan mencari data publik yang diambil pada situs online *data.world*, sebuah situs yang menyediakan data yang dapat ditemukan, dipahami dan digunakan oleh siapa saja. dengan link *https://data.world/data-society/used-cars-data* [10]. Data yang diambil merupakan data iklan penjualan mobil bekas dengan jumlah 371.539 *record* dan 20 attribut diambil dari *Ebay-Kleinanzeigen*, sebuah situs *online* penjualan mobil di Jerman. Spesifikasi data yang digunakan dalam penelitian ini ditunjukkan pada tabel 3.1 dan 3.2 sebagai berikut :

Tabel 3.3 Attribut data bertipe Kategori

No	Attribut	Type Data	Missing Value
1	dateCrawled	Categorical	0
2	name	Categorical	0
3	seller	Categorical	1
4	offerType	Categorical	1
5	abtest	Categorical	1
6	vehicleType	Categorical	37870
7	gearbox	Categorical	20210
8	model	Categorical	20485
9	kilometer	Categorical	1
10	fuelType	Categorical	33388
11	brand	Categorical	2
12	notRepairedDamage	Categorical	72062
13	dateCreated	Categorical	2
14	lastSeen	Categorical	2

Tabel 3.4. Attribut data bertipe Numerik

No	Attribut	Type Data	Missing Value	Min	Max	Kemiringan
1	yearOfRegistration	Numerical	2	1000	9.999	72,134
2	powerPS	Numerical	1	0	20.000	58,200
3	postalCode	Numerical	2	1067	99.998	0,0637
4	nrOfPictures	Numerical	2	0	0	0,0
5	price	Numerical	1	0	2.147.483.647	578,063
6	monthOfRegistration	Numerical	2	0	12	0,0790

Pada tabel 3.3 dan 3.4 merupakan atribut dari data penjualan mobil bekas yang sudah dipisahkan berdasaran tipe data,terdapat 14 atribut memiliki tipe kategori dan 6 atribut bertipe numerik. Hampir semua atribut terdapat *missing values* dan pada atribut tipe numerik rata-rata mengalami permasalahan pada distribusi data yang tidak wajar atau data *outlier*. Oleh karena itu, akan dilakukan *preprocessing* data untuk menyiapkan data menjadi siap diproses oleh algoritma *machine learning* dan agar mendapatkan performa yang terbaik.

3.3 Instrumen Penelitian

Berdasarkan permasalahan yang telah dijelaskan pada bab sebelumnya, maka Instrumen penelitian yang dibutuhkan untuk penelitian ini adalah :

1. Penelitian menggunakan data publik/sekunder yang diperoleh dari situs *online data.world*, sebuah situs yang menyediakan data yang dapat ditemukan, dipahami dan digunakan oleh siapa saja. dengan link *https://data.world/data-society/used-cars-data* [10]. Data disajikan dalam bentuk tabel sebanyak 20 atribut dan 371539 *record*.
2. Algoritma yang digunakan untuk penelitian yaitu *XGBoost* dengan *GridSearchCV Cross Validation*.
3. Teknik pembersihan data dengan *replace missing value*.
4. *Feature selection* dengan *matrix correlation*. *Matrix correlation* digunakan untuk mencari seluruh hubungan antar atribut numerik dalam dataset.
5. Teknik *percentiles* dan *Log transformation* untuk menangani data *outlier*.
6. Data *encoding* untuk merubah data string ke data numerik.
7. Metode *normalization* dengan *RobustScaler*
8. Perangkat lunak yang digunakan untuk penelitian yaitu *Python* yang dijalankan pada *google colab*.

3.4 Pemisahan Dataset *Training* dan *Testing*

Dalam membuat suatu model m*achine Learning*, data dibagi menjadi data *training* dan data *testing*. *Machine* harus tahu yang mana akan dijadikan set data untuk dicapai / dilampaui, dan yang mana akan dijadikan set data untuk mencapai / melampaui tujuan penelitian. Dengan melatih model dengan data *training* dan menilai kinerja klasifikasi atau prediksi dengan data *testing* dapat diketahui model kinerja tersebut handal atau tidak.

Pada penelitian ini, data *training* dan data *testing* dipisah dengan *Cross Validation* dengan k sebesar 10. Jumlah k 10 tersebut direkomendasikan untuk pemilihan model terbaik karena mampu mengurangi peluang terjadinya bias dibandingkan dengan *cv* biasa seperti *Hold out method* dan *Leave one out*. Dimana 10 *fold cv* akan membagi data ke dalam partisi k sebesar 10 dengan pengambilan jumlah sampel yang sama sehingga terdapat 10 subset data yang akan mengevaluasi kinerja

dari model. Untuk masing-masing dari 10 subset data tersebut, *cv* akan menggunakan sembilan *fold* untuk pelatihan dan satu *fold* untuk pengujian [50].

3.5 Evaluasi dan Validasi Hasil

Pada tahap ini akan dilakukan proses pengujian metode dan model yang digunakan dengan mengevaluasi model *XGBoost* dengan *GridSearch Cross Validation* dan sebagai pembanding performa menggunakan *XGBoost* tanpa *GridSearch Cross Validation*, serta proses *preprocessing* yang menghasilkan model prediksi yang dianggap paling optimal dalam prediksi harga mobil bekas berdasarkan nilai *MSE, MAE* terkecil dan *R2 score* tertinggi.

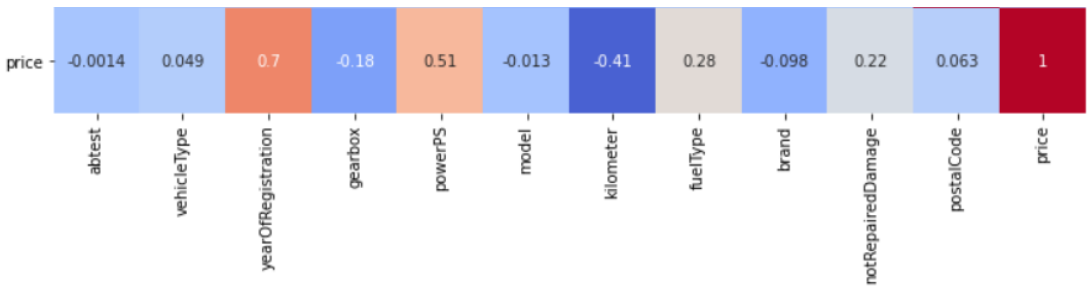
BAB IV
HASIL PENELITIAN DAN PEMBAHASAN

4.1. Eksperimen dan Pengujian Model

Pada tahapan ini terdiri atas beberapa bagian eksperimen yaitu, eksperimen pertama digunakan untuk mencari *baseline* model *XGBoost* untuk prediksi harga mobil bekas. Dilakukan beberapa *preprocessing* agar data dapat diproses oleh model XGBoost. Kemudian eksperimen kedua digunakan untuk membandingkan performa model prediksi harga mobil bekas antara model *XGBoost* tanpa *Hyperparameter Tuning* dan dengan *Hyperparameter Tuning*. Dilakukan beberapa proses *preprocessing* data seperti seleksi atribut untuk prediksi harga mobil, kemudian menangani masalah data yang memiliki pencilan atau *outlier* dan normalisasi data.

4.1.1. Seleksi Atribut

Pada tahapan ini dilakukan proses seleksi atribut yang akan digunakan untuk prediksi harga mobil bekas, dari 20 atribut yang dimiliki, penulis mengambil sebanyak 12 atribut dan menghapus sebanyak 8 atribut seperti atribut *dateCrawled, monthOfRegistration, dateCreated, lastSeen, name, seller, offerType, nrOfPictures*. Berikut adalah gambar untuk data korelas dari 12 atribut yang dipilih terhadap atribut target yaitu *price*.



Gambar 4.1 Korelasi atribut terhadap atribut target

Merujuk dari penelitian sebelumnya, pada gambar 4.1 merupakan atribut yang digunakan untuk prediksi harga mobil bekas, bahwa untuk atribut tanggal dan bulan dihapus karena kurang berpengaruh terhadap prediksi harga, dengan demikian atribut tersebut dihapus untuk meningkatkan performa model. Selain itu, atribut seperti nama dihapus karena memiliki varian data yang tinggi dan akan mengurangi performa dari model. Kemudian yang terakhir atribut *seller, offerType* dan *nrOfPictures* juga dihapus

karena memiliki jumlah nilai-nilai yang tidak seimbang, sehingga akan berdampak pada penurunan performa dari model prediksi.

Tabel 4.1 Standar Deviasi masing-masing atribut

No	Attribut	Standar Deviasi
1	abtest	0.49
2	vehicleType	1.90
3	yearOfRegistration	4.84
4	gearbox	0.45
5	powerPS	46.46
6	model	71.08
7	kilometer	38031.75
8	fuelType	1.21
9	brand	13.39
10	notRepairedDamage	0.68
11	postalCode	25608.86
12	price	0.94

Tabel 4.1 merupakan nilai standar deviasi masing-masing atribut, dari 12 atribut yang dipilih, kemudian dilakukan beberapa *preprocessing* data dengan teknik tertentu maka diperoleh nilai standar deviasi seperti terlihat pada tabel. Standar deviasi tersebut dapat dijadikan tolak ukur dari jumlah variasi atau sebaran dari sejumlah nilai data yang mana jika nilai standar deviasi semakin kecil maka penyebaran nilai atribut tersebut mendekati rata-rata, namun jika nilai standar deviasi tersebut semakin tinggi berarti semakin tinggi atau semakin jauh rentang variasi datanya dari rata-rata.

4.1.2. Menangani Pencilan atau *Outlier*

Pada tahapan ini dilakukan beberapa proses yaitu deteksi *outlier* dengan *percentiles*, *Log Transformation*, dan *data Normalization*. Berikut penjelasan mengenai beberapa proses yang telah dilakukan oleh penulis:

1. Deteksi *Outlier*

   Deteksi *outlier* disini menggunakan teknik *percentiles* yaitu dengan mengasumsikan persentase nilai bawah dan nilai atas sebagai *outlier*. Untuk persentase nilai *percentiles* diambil 0,1 persen dari nilai bawah sebagai *outlier* dan 3 persen dari nilai atas sebagai *outlier*. Teknik persentil disini diterapkan terhadap 3 atribut yaitu *price*, *powerPs* dan *yearOfRegistration*. Hal tersebut

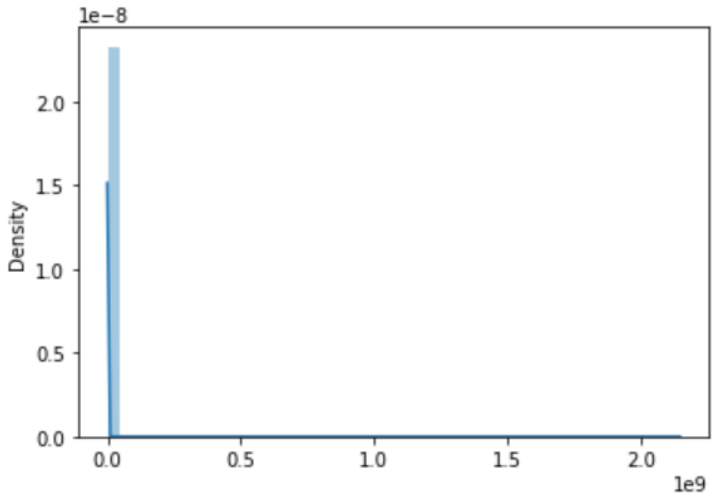
dilakukan karena masing-masing atribut memiliki distribusi penyebaran data (*skew*) yang cukup tinggi jika dibandingkan dengan atribut yang lainnya.

Tabel 4.2 *Range* nilai setelah deteksi *outlier* dengan persentil

	Nilai bawah	**Nilai atas**
price	500	24500
powerPs	54	265
yearOfRegistration	1995	2017

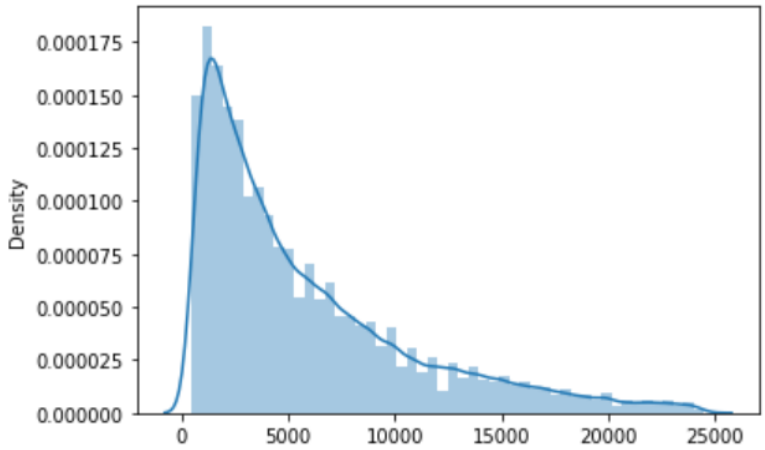
Pada tabel 4.2 diatas, menunjukkan hasil dari proses deteksi *outlier* menggunakan teknik persentil, ketiga atribut masing-masing memiliki *range* baru dari distribusi datanya. Atribut *price* terlihat memiliki *range* nilai antara 500 sampai dengan 24.500 yang distribusi semula dari harga 0 sampai dengan 2.147.483.647, atribut *powerPs* memiliki *range* nilai antara 54 sampai dengan 265 yang distribusi semula dari nilai 0 sampai dengan 20.000 dan atribut *yearOfRegistration* memiliki *range* nilai antara tahun 1995 sampai dengan 2017 yang distribusi sebelumnya mulai dari tahun 1000 sampai dengan tahun 9.999. Data yang tersisa setelah dilakukan proses deteksi *outlier* dengan persentil yaitu sebesar 242.296 dengan atribut sebanyak 12.

Berikut adalah gambar dari distribusi kemiringan *price, powerPs,* dan *yearOfRegistration* setelah dilakukan teknik *percentiles* terlihat pada gambar 4.3, 4.4 dan 4.5 :



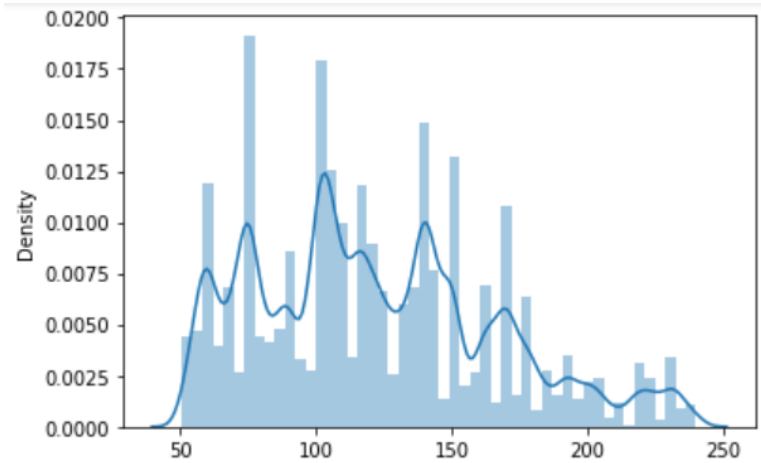
Gambar 4.2 Distribusi kemiringan harga sebelum deteksi *outlier*

Pada gambar 4.2 merupakan visualisasi dari distribusi kemiringan harga sebelum dilakukan proses deteksi *outlier* dengan *percentiles*. Dari data sebanyak 371.539 record tersebut memiliki penyebaran harga dikisaran mendekati 0 namun terdapat juga harga yang sangat jauh dari penyebaran data lainnya yaitu sebesar 2.147.483.647.



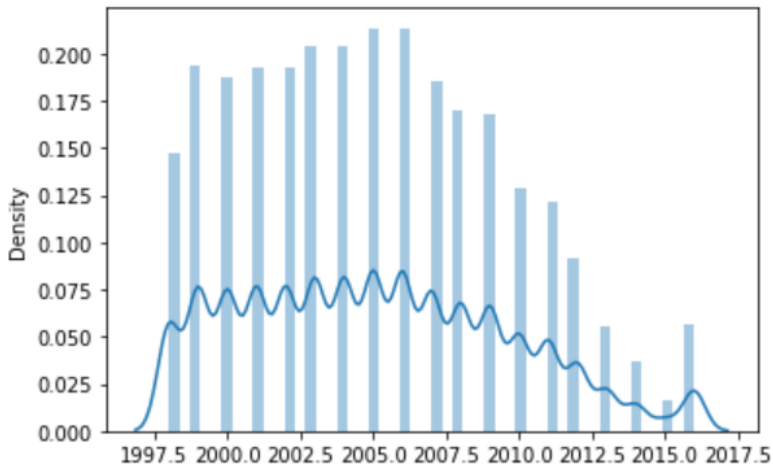
Gambar 4.3 Distribusi kemiringan harga setelah deteksi *outlier*

Pada gambar 4.3 merupakan visualisasi dari distribusi kemiringan harga setelah dilakukan proses deteksi *outlier* dengan *percentiles*. Nampak masih terjadi kemiringan lonceng ke kanan, sehingga dapat disimpulkan bahwa masih terjadi distribusi harga yang belum normal dari kumpulan data yang ada.



Gambar 4.4 Distribusi kemiringan *powerPs* setelah deteksi *outlier*

Pada gambar 4.4 merupakan visualisasi dari distribusi kemiringan *powerPs* setelah dilakukan proses deteksi *outlier* dengan *percentiles*. Nampak sudah terjadi distribusi kemiringan *powerPs* yang cukup baik dibandingkan dengan distribusi sebelumnya.

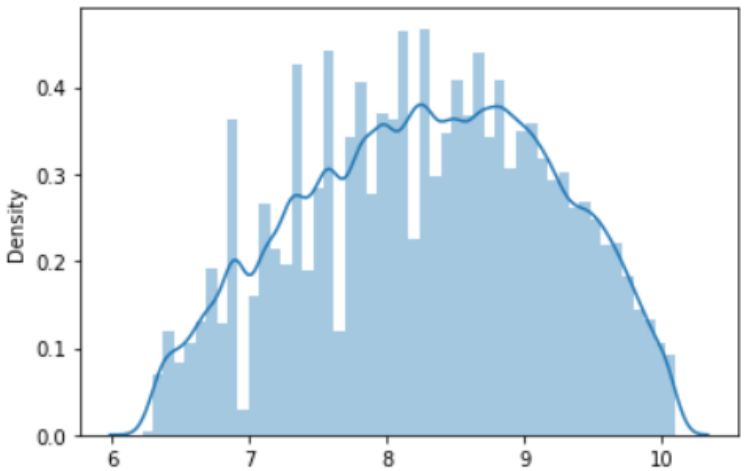


Gambar 4.5 Distribusi kemiringan *yearOfRegistration* setelah deteksi *outlier*

Pada gambar 4.5 merupakan visualisasi dari distribusi kemiringan *yearOfRegistration* setelah dilakukan proses deteksi *outlier* dengan *percentiles*. Nampak sudah terjadi distribusi kemiringan *powerPs* yang baik dibandingkan dengan distribusi kemiringan dari atribut *price* dan *powerPs*.

2. *Log Transformation*

Setelah dilakukan proses deteksi *outlier* dengan teknik *percentiles*, didapatkan bahwa masih terjadi kemiringan kurva lonceng yang terjadi pada atribut *price*, sehingga masih perlu dilakukan proses penormalan data untuk atribut *price*. Proses yang dilakukan yaitu dengan menerapkan *transformation log*. *transformation log* disini dilakukan untuk melihat seberapa besar pengaruh performa yang diberikan untuk menangani distribusi kemiringan data, hasil yang diperoleh terlihat pada gambar 4.6 sebagai berikut:



Gambar 4.6 Distribusi kemiringan harga setelah di *transformation log*

Pada gambar 4.6 terlihat bahwa distribusi kemiringan atau kurva lonceng dari atribut harga setelah dilakukan *log transformation* menjadi lebih baik, yang awalnya terjadi kemiringan ke kanan sebesar 1,36 *skew*. Namun setelah dilakukan *log transformation*, kemiringan kurva menjadi sebesar -0,06 *skew* dan memiliki *range* nilai antara 6 sampai dengan 10.

3. Data Normalization

Pada tahap selanjutnya dilakukan proses normalisasi data dengan menggunakan metode *RobustScaler*. *RobustScaler* memproses data dengan menggunakan nilai median dan kuartil, bukan *mean* dan varians. Hal ini membuat *scaler* mengabaikan titik data yang sangat berbeda dari yang lain sehingga *RobustScaler* ini cocok digunakan untuk menangani distribusi data yang besar atau disebut dengan *outlier*.

4.1.3. Membandingkan Performa Model

Setelah dilakukan *preprocessing* data seperti mengganti data yang hilang pada atribut bertipe kategori, menghapus data yang hilang pada atribut bertipe numerik dan menangani data yang memiliki distribusi penyebaran tinggi atau sering disebut *outlier*. Kemudian, dilakukan beberapa percobaan dengan algoritma *XGBoost*, dan algoritma *Linear Regression* untuk membandingkan performa model *XGBoost* sebelum dilakukan proses *hyperparameter tuning*. Dari beberapa eksperimen yang dilakukan, diperoleh hasil sebagai berikut :

1. Baseline *XGBoost* dan *Linear Regression*

   Pada eksperimen pertama dengan percobaan menggunakan baseline *XGBoost* dan *Linear Regression*, yang dilakukan adalah menghapus data *NaN* atau *missing value*. Hal tersebut dilakukan karena model tidak dapat memproses data yang masih memiliki nilai kosong. Setelah itu dilakukan proses transformasi log agar menjadi pembanding yang sama dengan eksperimen ke dua dan transformasi tipe data kategori menjadi tipe numerik, hal tersebut dilakukan karena model hanya mampu memproses data bertipe numerik. Setelah data siap diproses, tersisa sebesar 260.965 field dan 20 atribut dipakai semua. Berikut adalah hasil menggunakan *baseline XGBoost* dan *baseline Linear Regression* yang dimasukkan dalam tabel 4.3.

Tabel 4.3 Performa *Baseline XGBoost* dan *Linear Regression*

Model	R2	MAE	MSE	MAPE	MPE
Baseline XGB	0.80%	0.32	0.27	358.5	-333.3
Baseline LR	0.50%	0.57	0.69	19692.2	-19654.8

Tabel 4.3 merupakan hasil dari performa *baseline XGBoost* dan *Linear Regression* tanpa menggunakan parameter, masing-masing algoritma memiliki nilai *R2* sebesar 0.80%, dan 0.50%. kemudian nilai *MAE* sebesar 0.32, dan 0.57 dilanjutkan nilai *MSE* sebesar 0.27 dan 0.69. Selain itu, ditunjukkan pula hasil nilai *error* dengan *MAPE* dan *MPE* yaitu persentase kesalahan rata-rata secara mutlak, dengan *MAPE* dan *MPE* diperoleh hasil dengan kesalahan lebih kecil pada algoritma *XGBoost*. Namun dari hasil eksperimen tersebut menunjukkan bahwa performa *baseline* dari algoritma *XGBoost* dan *Linear Regression* masih buruk untuk digunakan dalam pemodelan prediksi harga mobil bekas, karena akurasi *R2* dikatakan baik jika mendekati angka 1 dan nilai *error* yang buruk jika angka mendekati 0. Hasil terbaik dari *baseline XGBoost* tersebut dirasa masih dapat dilakukan optimalisasi performa.

2. *XGBoost Hyperparameter Tuning GridSearchCV*

Pada eksperimen kedua, penulis membangun model *XGBoost* tanpa *Hyperparameter Tuning GridSearchCV* kemudian akan dibandingkan dengan model *XGBoost Hyperparameter Tuning GridSearchCV*. Pada model *XGBoost* tanpa *Hyperparameter Tuning*, data diproses tanpa menggunakan parameter apapun yang artinya algoritma tersebut akan secara otomatis menggunakan parameter *default* dari *XGBoost*. Setelah mendapatkan hasil performa prediksi dari model *XGBoost* tanpa *Hyperparameter Tuning GridSearchCV* kemudian dilakukan eksperimen dengan model *XGBoost Hyperparameter Tuning GridSearchCV* untuk membanding hasil performa. Dari eksperimen yang dilakukan diperoleh hasil seperti pada tabel berikut :

Tabel 4.4 *Hyperparameter Tuning XGBoost*

max depth	**learning rate**	**min child weight**
4,7, 9, 11	0.01, 0.1	1, 2, 3,4

Pada tabel 4.4 merupakan *hyperparameter tuning XGBoost* beserta parameter tambahan lainnya, sebanyak 3 parameter dilakukan proses *tuning*, yaitu terhadap parameter *max_depth, learning_rate* dan *min_child_weight*. Dengan memberikan *range* nilai pada masing-masing parameter, maka teknik *GridSearch* disini akan bekerja mencari model terbaik untuk prediksi harga mobil bekas. Dari masing-masing nilai yang diberikan, diperoleh model terbaik dengan parameter *max_depth* sebesar 11, *learning_rate* terbaik yaitu 0.1 dan *min_child_weight* sebesar 3. Dari ketiga kombinasi parameter tersebut ditambah juga dengan beberapa parameter lain seperti pada tabel 4.5 berikut ini :

Tabel 4.5 Parameter tambahan *XGBoost*

n_estimators	**subsample**	**objective**	**colsample_ bytree**	**n_jobs**
500	0,8	reg:linear	0,8	**1**

Pada tabel 4.5 merupakan parameter-parameter lain beserta nilainya yang tidak dituning. Dengan memberikan parameter-parameter tersebut terjadi peningkatan performa yang lebih baik terhadap nilai *R2* skor, *MAE* dan *MSE* untuk prediksi harga mobil bekas jika dibandingkan dengan hasil yang diberikan oleh *baseline XGBoost*. Hasil tersebut merupakan performa terbaik dari model *XGBoost Hyperparameter Tuning GridSearchCV*.

Tabel 4.6 Hasil performa dari masing-masing model

Model	**R2**	**MAE**	**MSE**	**MAPE**	**MPE**
Baseline XGB	0.8%	0.32	0.27	358.5	-333.3
XGBoost tanpa *Hyperparameter Tuning*	0,85%	0,27	0,13	29.73	-7.13
XGBoost *Hyperparameter Tuning*	0,89%	0,21	0,09	23.31	-4.96

Pada tabel 4.6 menunjukkan hasil performa dari masing-masing eksperimen yang telah dilakukan, terlihat bahwa model *XGBoos*t tanpa *Hyperparameter Tuning* menghasilkan nilai *R2* skor sebesar 0,84% yang mana terjadi peningkatan performa yang lebih baik sebesar 0,04% ketika menggunakan model *XGBoost Hyperparameter Tuning*, sedangkan nilai *error* yang diperoleh dengan model *XGBoost* tanpa *Hyperparameter Tuning* sebesar *MAE*=0,27 dan *MSE*=0,13, Nilai *error* tersebut menurun mendekati angka 0 ketika menggunakan model *XGBoost Hyperparameter Tuning*. Hasil tersebut membuktikan bahwa, dengan memberikan *hyperparameter tuning* terhadap algoritma *XGBoost* mampu menaikkan performa prediksi harga mobil bekas.

3. *Linear Regression*

Pada tahapan ini dilakukan eksperimen dengan menggunakan algoritma *Linear Regression*. Tujuan dari eksperimen ini adalah untuk membandingkan hasil performa *XGBoost* dengan performa algoritma lain. Pada tahap ini dilakukan uji coba dengan mencari performa *baseline* dari *Linear Regression* dan juga performa ketika data sudah dilakukan *preprocessing* data. Berikut adalah tabel hasil dari perbandingan algoritma *XGBoost* dan *Linear Regression* :

Tabel 4.7 Hasil performa dari *Linear Regression*

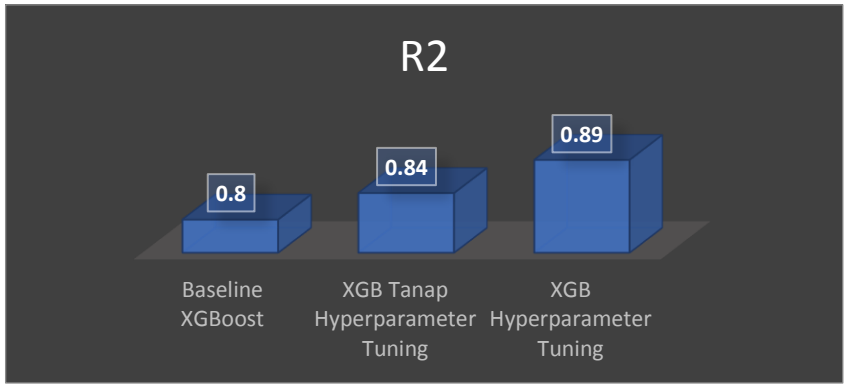
Model	R2	MAE	MSE	MAPE	MPE
Baseline XGB	0.8%	0.32	0.27	358.5	-333.3
Baseline LR	0.5%	0.57	0.69	19692.2	-19654.8
LR	0,68%	0,38	0,27	47.5	-18.29
XGBoost	0,84%	0,27	0,13	29.11	-7.13
XGBoost GridSearch CV	0,89%	0,21	0,09	23.31	-4.96

Pada tabel 4.7 merupakan hasil dari perbandingan performa *XGBoost* dan *Linear Regression*. Terlihat bahwa dengan menggunakan algoritma *Linear Regression*, hasil dari *baseline* masih jauh lebih buruk jika dibandingkan dengan *baseline XGBoost* dan hasil performa *Linear Regression* ketika sudah dilakukan *preprocessing* data menunjukkan bahwa jauh lebih buruk dibandingkan dengan

XGBoost. Sehingga dapat disimpulkan bahwa performa untuk *XGBoost* lebih baik jika dibandingkan dengan performa *Linear Regression*.

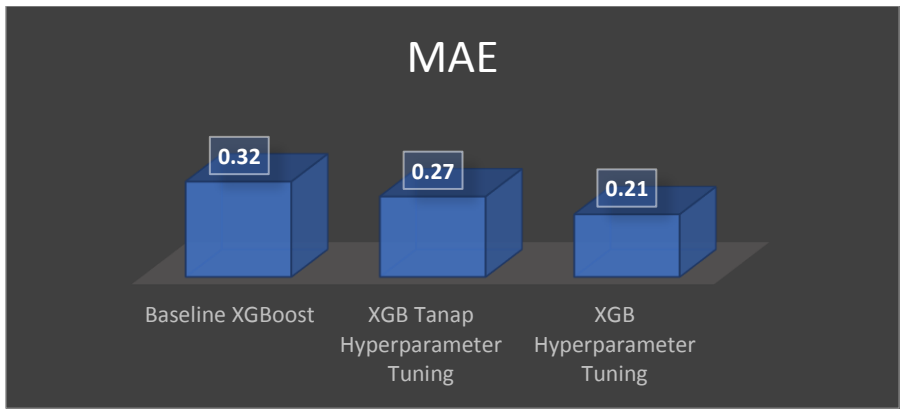
4.1.4. Evaluasi dan Validasi Model

Dalam evaluasi dan validasi hasil ini dilakukan perbandingan nilai *R2*, *MAE* dan *MSE* berdasarkan beberapa eksperimen yang telah dilakukan terhadap algoritma *XGBoost* dari tahap *preprocessing* data, normalisasi dan penyetingan beberapa parameter pada setiap model. Berikut ini adalah grafik yang ditunjukkan dari hasil *R2* skor *XGBoost*:



Gambar 4.7 Perbandingan Akurasi validasi *R2* skor

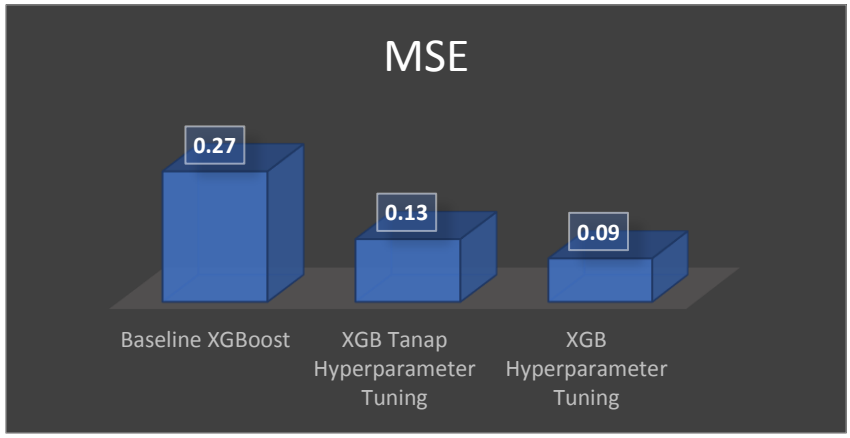
Pada gambar 4.7 merupakan hasil perbandingan akurasi *R2* skor antara *baseline XGBoost*, *XGBoost* tanpa *Hyperparameter Tuning GridSearch*CV dan dengan *Hyperparameter Tuning GridSearchCV*. Hasil menunjukkan akurasi terbaik diperoleh dengan metode *Tuning* yaitu sebesar 0.89% yang mana hasil tersebut meningkat sebesar 0.05% ketika model diberikan parameter tambahan.



Gambar 4.8 Perbandingan Akurasi Validasi *MAE*

Pada gambar 4.8 merupakan hasil perbandingan dari nilai *error* menggunakan pengukuran *MAE*. Hasil menunjukkan bahwa *error* terkecil diperoleh dengan

menggunakan metode *XGBoost Hyperparameter Tuning GridsearchCV* dengan nilai sebesar 0,21. Hasil tersebut dapat disimpulkan bahwa nilai *error* semakin mengecil ketika dilakukan proses *tuning* terhadap parameter yang diberikan dan performa prediksi harga mobil juga semakin baik.



Gambar 4.9 Perbandingan Validasi *MSE*

Pada gambar 4.9 merupakan hasil perbandingan dari nilai *error* menggunakan pengukuran *MSE*. Hasil menunjukkan bahwa *error* terkecil diperoleh menggunakan metode *XGBoost Hyperparameter Tuning GridsearchCV* dengan nilai sebesar 0,09 dan diikuti dengan nilai *error* terkecil selanjutnya yaitu dengan metode *XGBoost* tanpa *Hyperparameter Tuning* sebesar 0,13.

Dari ketiga hasil tersebut, membuktikan bahwa dengan melakukan *preprocessing* secara tepat serta memberikan metode *tuning* terhadap beberapa parameter pada algoritma *XGBoost* mampu memberikan dampak yang cukup baik. Terjadi peningkatan performa terhadap akurasi *R2* skor dan nilai *error* semakin menurun mendekati 0.

4.2. Pembahasan

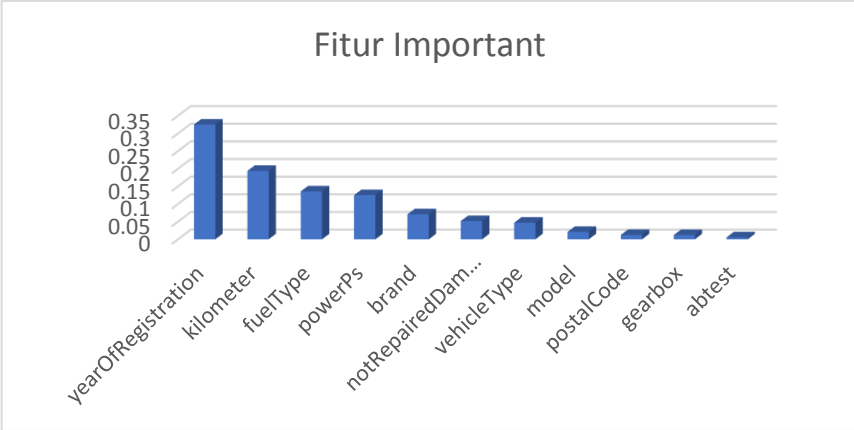
Dari beberapa eksperimen yang telah dilakukan terhadap algoritma *XGBoost* diperoleh hasil bahwa performa terbaik untuk memprediksi harga mobil bekas ditunjukkan pada eksperimen *XGBoost* dengan metode *Hyperparameter Tuning GridSearch* yang mana dengan metode tersebut diperoleh hasil terbaik dengan penyetelan beberapa *hyperparameter tuning*. Dengan nilai akurasi sebesar 0.89%,

nilai *error MAE* sebesar 0,21 dan nilai *error MSE* sebesar 0,09 menunjukkan bahwa performa untuk prediksi harga mobil bekas dikatakan sangat baik. Berikut tabel dari beberapa contoh hasil prediksi harga asli mobil bekas dengan harga prediksi dari model yang diperoleh.

Tabel 4.8 Prediksi harga mobil

No	Actual_price	predict_price	Diff
1	16990	17837.6	-847.64
2	8950	9571.2	-621.2
3	2600	1786.37	813.632
4	6100	4518.29	1581.71
5	3000	2675.23	324.772
6	6900	8292.89	-1392.9
7	17500	10692.4	6807.58
8	1100	1022.36	77.6388
9	4500	3164.91	1335.09
10	2450	2426.72	23.2791

Tabel 4.8 merupakan hasil dari prediksi harga mobil bekas yang mana atribut actual *price* adalah harga asli dari mobil bekas dan *predict price* adalah harga prediksi dari model yang dibangun serta atribut *Diff* adalah selisih dari harga asli dan harga prediksi harga mobil bekas. Terdapat 10 contoh harga mobil bekas yang diprediksi dan masing-masing memiliki selisih yang tidak terlalu tinggi. Hasil menunjukkan bahwa terdapat prediksi harga yang semakin tinggi/mahal dan ada yang semakin menurun/murah, hal tersebut dipengaruhi oleh beberapa atribut yang terdapat pada data mobil yang ada. Seperti yang ditunjukkan pada tabel 4.8 merupakan nilai *feature importance* dari masing-masing atribut dan nilai korelasi masing-masing atribut terhadap atribut target yaitu *price*.



Gambar 4.10 Nilai *Feature Importance*

Gambar 4.10 merupakan diagram dari nilai *Feature Importance*, yang mana atribut *yearOfRegistration* adalah atribut yang memiliki nilai *Feature Importance* paling tinggi, hal tersebut menunjukkan bahwa tahun pembuatan mobil paling berpengaruh terhadap prediksi harga mobil bekas. Dilanjutkan atribut kedua yaitu kilometer atau jarak tempuh mobil merupakan atribut kedua yang paling berpengaruh terhadap prediksi harga mobil. Selain itu, atribut *fuelType* juga merupakan atribut yang berpengaruh ketiga terhadap prediksi harga mobil bekas. Dilanjutkan atribut berpengaruh selanjutnya yaitu *powerPs, brand, notRepairedDamage, vehicleType, model, postalcode, gearbox* dan terakhir yaitu atribut *abtest*.

Tabel 4.9 *Feature Importance*

Atribut	**Nilai Fitur Important**	**Nilai Korelasi terhadap target**
yearOfRegistration	0.325132	0,7
kilometer	0.193824	-0,41
fuelType	0.135534	0,28
powerPs	0.125049	0,51
brand	0.070909	-0,098
notRepairedDamage	0.051555	0,22
vehicleType	0.047001	0,049
model	0.020958	-0,013
postalCode	0.012522	0,063

gearbox	0.012015	-0,18
abtest	0.005501	-0,0014

Tabel 4.9 merupakan atribut yang digunakan beserta nilai *feature importance* dan korelasinya terhadap atribut *price*. Dari tabel tersebut dapat dilihat bahwa 3 atribut yang paling berpengaruh yaitu *yearOfRegistration*, kilometer, dan *powerPs*. Masing-masing memiliki hubungan atau korelasi negatif dan korelasi positif. Atribut *yearOfRegistration*, dan *powerPs* memiliki korelasi yang positif terhadap prediksi harga mobil bekas, yang berarti semakin baru tahun pembuatan mobil, dan semakin tinggi tenaga yang dimiliki mobil akan semakin tinggi harga yang dapat ditawarkan, sebaliknya jika atribut kilometer memiliki korelasi yang negatif terhadap prediksi harga mobil bekas, karena semakin tinggi nilai kilometer atau jarak tempuh mobil maka akan semakin menurun harga mobil yang ditawarkan.

BAB V
PENUTUP

5.1 Kesimpulan

Penelitian ini dilakukan untuk mendapatkan model terbaik dari algoritma *XGBoost* sehingga mampu memberikan performa hasil prediksi yang akurat terhadap harga mobil bekas. Selain itu, untuk mencari metode *preprocessing* yang tepat agar memberikan dampak paling efektif untuk prediksi harga mobil bekas serta menganalisa hubungan antar atribut yang berpengaruh terhadap prediksi harga mobil bekas.

Berdasarkan penelitian yang sudah penulis lakukan, maka dapat disimpulkan sebagai berikut :

1. Penentuan hasil prediksi yang baik dari suatu data ditentukan pada pemilihan model dan parameter yang baik pula. *Hyperparameter Tuning GridSearchCV* terbukti memberikan kemudahan dalam uji coba setiap model dan parameter dalam algoritma *XGBoost* tanpa harus melakukan validasi satu persatu. *GridSearchCV* mampu menemukan model terbaik dari algoritma *XGBoost* dengan *Hyperparameter tuning* yang telah di *setting* nilainya. Model terbaik *XGBoost* yang ditemukan yaitu dengan nilai parameter *CV* 10 dan *max_depth* sebesar 11 kemudian dengan *learning rate* sebesar 0.1 dan *min_child_weight* sebesar 3 menunjukkan hasil terbaiknya dalam memprediksi harga mobil bekas.
2. Deteksi *outlier* dengan *percentiles* dan *Log transformation* mampu mengurangi penyebaran distribusi data yang terlalu jauh sehingga tingkat kemiringan distribusi data menjadi mendekati 0 dan mampu meningkatkan performa prediksi harga mobil bekas secara signifikan.
3. Hasil evaluasi dan validasi yang dilakukan dengan menggunakan *Mean Square Error, Mean Absolute Error* & *R2 Score* menunjukkan bahwa model yang dibangun menggunakan algoritma *XGBoost* dengan *Hyperparameter Tuning GridSearchCV* mampu menemukan secara otomatis model dengan performa yang lebih baik dibandingkan dengan *XGBoost* tanpa

Hyperparameter Tuning GridSearchCV dengan nilai eror *MAE* = 0.21, *MSE*= 0.09, dan *R2* = 0.89.

4. Dari penelitian yang telah dilakukan, penulis dapat melihat bahwa fitur yang paling berpengaruh terhadap model adalah fitur tahun pembuatan mobil (*yearOfRegistration*). Fitur *yearOfRegistration* mempengaruhi prediksi harga mobil bekas, semakin baru tahun pembuatan model maka harga mobil bekas juga akan semakin naik. Sedangkan fitur kilometer atau jarak tempuh mobil juga menjadi fitur berpengaruh terhadap prediksi harga mobil. Namun, korelasi kilometer dengan harga menjadi korelasi *negative* karena semakin tinggi jarak tempuh mobil semakin turun harga mobil tersebut.

5.2 Saran

Berdasarkan hasil penelitian yang telah dilakukan penulis, menunjukkan bahwa algoritma *XGBoost* dengan *Hyperparameter Tuning* dan proses *preprocessing* yang tepat sudah baik untuk memprediksi harga mobil bekas, akan tetapi untuk penelitian selanjutnya agar dapat ditingkatkan dan dikembangkan kembali, berikut adalah saran yang diusulkan :

1 Menyetel kembali nilai dari parameter yang digunakan untuk setiap model agar mendapatkan performa yang lebih baik lagi dalam memprediksi harga mobil bekas.

2 Menggunakan proses *preprocessing* data yang lain seperti metode untuk menangani data yang *outlier* dan normalisasi data yang lain.

3 Melakukan pengembangan dengan *feature selection* lain untuk memilih atribut yang berpengaruh kuat terhadap model sehingga atribut yang dipakai hanya sedikit akan tetapi tidak mengurangi performa dari model yang dibuat.

DAFTAR PUSTAKA

[1] TMFWAI, *The future of work in the automotive industry: The need to invest in people's capabilities and decent and sustainable work*, no. May. 2020.

[2] Kontan.co.id, "CARRO raih hasil penjualan mobil bekas yang positif di kuartal I-2021," 2021. http://industri.kontan.co.id/news/carro-raih-hasil-penjualan-mobil-bekas-yang-positif-di-kuartal-i-2021 (accessed Jul. 31, 2021).

[3] B. Kriswantara, Kurniawati, and H. F. Pardede, "PREDIKSI HARGA MOBIL BEKAS DENGAN MACHINE LEARNING," *Syntax Lit. J. Ilm. Indones.*, vol. 6, no. 5, p. 6, 2021, [Online]. Available: http://dspace.ucuenca.edu.ec/bitstream/123456789/35612/1/Trabajo de Titulacion.pdf%0Ahttps://educacion.gob.ec/wp-content/uploads/downloads/2019/01/GUIA-METODOLOGICA-EF.pdf.

[4] E. Gegic, B. Isakovic, D. Keco, Z. Masetic, and J. Kevric, "Car price prediction using machine learning techniques," *TEM J.*, vol. 8, no. 1, pp. 113–118, 2019, doi: 10.18421/TEM81-16.

[5] A. D. Sharma and V. Sharma, "Used Car Price Prediction Using Linear Regression Model," *Int. Res. J. Mod. Eng. Technol. Sci.*, vol. 2, no. 11, pp. 946–953, 2020, [Online]. Available: https://irjmets.com/rootaccess/forms/uploads/IRJMETS462275.pdf.

[6] K. Noor and S. Jan, "Vehicle Price Prediction System using Machine Learning Techniques," *Int. J. Comput. Appl.*, vol. 167, no. 9, pp. 27–31, 2017, doi: 10.5120/ijca2017914373.

[7] J. W. G. Putra, *Pengenalan Konsep Pembelajaran Mesin dan Deep Learning*, vol. 4. 2020.

[8] N. Monburinon, P. Chertchom, T. Kaewkiriya, S. Rungpheung, S. Buya, and P. Boonpou, "Prediction of prices for used car by using regression models," *Proc. 2018 5th Int. Conf. Bus. Ind. Res. Smart Te*

chnol. Next Gener. Information, Eng. Bus. Soc. Sci. ICBIR 2018, pp. 115–119, 2018, doi: 10.1109/ICBIR.2018.8391177.

[9] N. Pal, P. Arora, P. Kohli, D. Sundararaman, and S. S. Palakurthy, "How Much is my car worth? A methodology for predicting used cars' prices using random forest," *Futur. Inf. Commun. Conf.*, vol. 886, pp. 413–422, 2018, doi: 10.1007/978-3-030-03402-3_28.

[10] data society, "Used Cars Data," *https://data.world/*, 2016. https://data.world/data-society/used-cars-data (accessed Jul. 22, 2021).

[11] D. Kurniawan, *Pengenalan Machine Learning dengan Python*. PT Elex Media Komputindo, 2020.

[12] F. Rohmawati, M. G. Rohman, and S. Mujilahwati, "Sistem Prediksi Jumlah Pengunjung Wisata Wego Kec.Sugio Kab.Lamongan Menggunakan Metode Fuzzy Time Series," *Jouticla*, vol. 3, no. 2, 2017, doi: 10.30736/jti.v2i2.66.

[13] I. Marina and D. A. Lestari, "Pentingnya Data Deret Waktu dalam Melakukan Perencanaan Produksi (The Importance of Time Series Data in Production Planning)," *Semin. Nas. Multi Disiplin Ilmu dan Call Pap.*, no. 2011, pp. 582–589, 2017, [Online]. Available: https://www.unisbank.ac.id/ojs/index.php/sendi_u/article/view/5087.

[14] K. PENGETAHUAN, "Pengertian Prediksi," 2020. https://www.kanal.web.id/pengertian-prediksi (accessed Jul. 22, 2021).

[15] A. Géron, "Hands-On Machine Learning With Scikit-Learn & TensorFlow," in *O'Reilly Media, Inc*, vol. 53, no. 9, 2017, p. 564.

[16] B. Purnama, *Pengantar Machine Learning Konsep dan Praktikum Dengan Contoh Latihan Berbasis R dan Python*. Informatika Bandung, 2019.

[17] R. Primartha, *Algoritma Machine Learning*. Informatika Bandung, 2021.

[18] C. Tianqi and C. Guestrin, "XGBoost: A Scalable Tree Boosting System Tianqi," *J. Assoc. Physicians India*, no. 8, p. 665, 2016, doi: http://dx.doi.org/10.1145/2939672.2939785.

[19] M. Chen, Q. Liu, S. Chen, Y. Liu, C. H. Zhang, and R. Liu, "XGBoost-Based Algorithm Interpretation and Application on Post-Fault Transient Stability Status Prediction of Power System," *IEEE Access*, vol. 7, pp. 13149–13158, 2019, doi: 10.1109/ACCESS.2019.2893448.

[20] P. Bajaj, R. Ray, S. Shedge, S. Vidhate, and N. Shardoor, "SALES PREDICTION USING MACHINE LEARNING ALGORITHMS," *Int. Res. J. Eng. Technol.*, vol. 7, no. 6, pp. 3619–3625, 2020, doi: 10.1109/INCET49848.2020.9154130.

[21] A. Jain, "Complete Guide to Parameter Tuning in XGBoost with codes in Python," *Analytics Vidhya*, 2016. https://www.analyticsvidhya.com/blog/2016/03/complete-guide-parameter-tuning-xgboost-with-codes-python/ (accessed Jul. 24, 2021).

[22] C. Li, "Preprocessing Methods and Pipelines of Data Mining: An Overview," *Semin. DATA Min.*, no. June, pp. 1–7, 2019, [Online]. Available: http://arxiv.org/abs/1906.08510.

[23] S. Ramírez-Gallego, B. Krawczyk, S. García, M. Woźniak, and F. Herrera, "A survey on data preprocessing for data stream mining: Current status and future directions," *Neurocomputing*, vol. 239, pp. 39–57, 2017, doi: 10.1016/j.neucom.2017.01.078.

[24] S. García, J. Luengo, and F. Herrera, *Data Preprocessing in Data Mining*. 2015.

[25] O. Alghushairy, R. Alsini, T. Soule, and X. Ma, "A review of local outlier factor algorithms for outlier detection in big data streams," *Big Data Cogn. Comput.*, vol. 5, no. 1, pp. 1–24, 2021, doi: 10.3390/bdcc5010001.

[26] I. M. K. Karo, "Implementasi Metode XGBoost dan Feature Importance untuk Klasifikasi pada Kebakaran Hutan dan Lahan," *J. Softw. Eng. Inf. Commun. Technol.*, vol. 1, no. 1, pp. 10–16, 2020, [Online]. Available: https://ejournal.upi.edu/index.php/SEICT/article/view/29347/13697.

[27] J. Brownlee, “How to Scale Data With Outliers for Machine Learning,” 2020. https://machinelearningmastery.com/robust-scaler-transforms-for-machine-learning/ (accessed Jul. 23, 2021).

[28] C. FENG, “Log-transformation and its implications for data analysis,” *Shanghai Arch Psychiatry*, 2014. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4120293/ (accessed Jul. 23, 2021).

[29] L. Metcalf and W. Casey, “Introduction to data analysis,” 2016. https://www.sciencedirect.com/topics/computer-science/log-transformation (accessed Jul. 23, 2021).

[30] M. F. Nugroho and S. Wibowo, “Fitur Seleksi Forward Selection Untuk Menetukan Atribut Yang Berpengaruh Pada Klasifikasi Kelulusan Mahasiswa Fakultas Ilmu Komputer UNAKI Semarang Menggunakan Algoritma Naive Bayes,” *J. Inform. Upgris*, vol. 3, no. 1, pp. 63–70, 2017, doi: 10.26877/jiu.v3i1.1669.

[31] S. Putatunda and K. Rama, “A Modified Bayesian Optimization based Hyper-Parameter Tuning Approach for Extreme Gradient Boosting,” *Int. Conf. Inf. Process. Internet Things, ICINPRO 2019 - Proc.*, pp. 22–27, 2019, doi: 10.1109/ICInPro47689.2019.9092025.

[32] Y. Wang and S. Ni, “A XGBoost risk model via feature selection and Bayesian hyper- parameter optimization,” *Int. J. Database Manag. Syst. (IJDMS)*, vol. 1, no. 1, pp. 243–250, 2020, [Online]. Available: https://www.semanticscholar.org/paper/A-XGBoost-risk-model-via-feature-selection-and-Wang-Ni/b56eb420482d8cf63cea34a840803af32a727a3c.

[33] A. Razak and E. Riksakomara, “Peramalan Jumlah Produksi Ikan dengan Menggunakan Backpropagation Neural Network (Studi Kasus: UPTD Pelabuhan Perikanan Banjarmasin,” *J. Tek. ITS*, vol. 6, no. 1, pp. 142–148, 2017, doi: 10.12962/j23373539.v6i1.22129.

[34] A. Handayani, A. Jamal, and A. A. Septiandri, “Evaluasi Tiga Jenis Algoritme Berbasis Pembelajaran Mesin untuk Klasifikasi Jenis Tumor Payudara,” *J. Nas.*

Tek. Elektro dan Teknol. Inf., vol. 6, no. 4, 2017, doi: 10.22146/jnteti.v6i4.350.

[35] C. Spark, "Hyperparameter tuning in XGBoost," *cambridgespark*, 2017. https://blog.cambridgespark.com/hyperparameter-tuning-in-xgboost-4ff9100a3b2f (accessed Jul. 24, 2021).

[36] X. Developers, "XGBoost Parameters," *Xgboost*, 2020. https://xgboost.readthedocs.io/en/latest/parameter.html (accessed Jul. 24, 2021).

[37] G. S. K. Ranjan, A. Kumar Verma, and S. Radhika, "K-Nearest Neighbors and Grid Search CV Based Real Time Fault Monitoring System for Industries," *2019 IEEE 5th Int. Conf. Converg. Technol. I2CT 2019*, no. June 2020, 2019, doi: 10.1109/I2CT45611.2019.9033691.

[38] A. Wibowo, "10 FOLD-CROSS VALIDATION," *BINUS Higher Education*, 2017. https://mti.binus.ac.id/2017/11/24/10-fold-cross-validation/ (accessed Jul. 27, 2021).

[39] D. Satriani, L. U. Khasanah, and N. A. Rizki, "Penerapan Metode Grid-Search Dalam Menentukan Parameter Model Pertumbuhan Penduduk Di Kota Samarinda," *Pros. Semin. Nas. Mat. Stat. dan Apl.*, pp. 65–74, 2019, [Online]. Available: http://jurnal.fmipa.unmul.ac.id/index.php/SNMSA/article/view/528.

[40] A. Botchkarev, "Performance Metrics (Error Measures) in Machine Learning Regression, Forecasting and Prognostics: Properties and Typology," pp. 1–37, 2018, [Online]. Available: http://arxiv.org/abs/1809.03006.

[41] A. Y. Prathama, A. Aminullah, and A. Saputra, "Pendekatan ANN (Artificial Neural Network) Untuk Penentuan Prosentase Bobot Pekerjaan Dan Estimasi Nilai Pekerjaan Struktur Pada Rumah Sakit Pratama," *J. Teknosains*, vol. 7, no. 1, pp. 1–82, 2017, doi: 10.22146/teknosains.30139.

[42] O. Center, "Orange: Metric Evaluation Model," *OnnoCenterWiki*, 2020. https://lms.onnocenter.or.id/wiki/index.php/Orange:_Metric_Evaluation_Model (accessed Jul. 24, 2021).

[43] A. A. Suryanto and A. Muqtadir, “Penerapan Metode Mean Absolute Error (MEA) Dalam Algoritma Regresi Linear Untuk Prediksi Produksi Padi,” *Saintekbu J. Sains dan Teknol.*, vol. 11, no. 1, pp. 78–83, 2019, doi: 10.32764/saintekbu.v11i1.298.

[44] A. B. Santoso, “Apa perbedaaan R Squared, R squared adjusted, dan R Squared Predicted,” 2018. https://agungbudisantoso.com/apa-perbedaaan-r-squared-r-squared-adjusted-dan-r-squared-predicted/.

[45] O. Celik and U. Omer Osmanoglu, “Prediction of The Prices of Second-Hand Cars,” *Eur. J. Sci. Technol.*, no. 16, pp. 77–83, 2019, doi: 10.31590/ejosat.542884.

[46] P. Venkatasubbu and M. Ganesh, “Used Cars Price Prediction using Supervised Learning Techniques,” *Int. J. Eng. Adv. Technol.*, vol. 9, no. 1S3, pp. 216–223, 2019, doi: 10.35940/ijeat.a1042.1291s319.

[47] D. Berrar, “Cross-Validation,” *Encycl. Bioinforma. Comput. Biol.*, vol. 1, 2018, Accessed: Jul. 23, 2021. [Online]. Available: https://www.researchgate.net/.

[48] Sugiyono., *Metode Penelitian Kuantitatif Kualitatif Dan R&D*. Bandung: Alfabeta, 2013.

[49] A. Saifudin, “METODE DATA MINING UNTUK SELEKSI CALON MAHASISWA PADA PENERIMAAN MAHASISWA BARU DI UNIVERSITAS PAMULANG,” *J. Teknol.*, vol. 10, no. 1, pp. 25–36, 2018, [Online]. Available: https://www.academia.edu/35836119/Metode_Data_Mining_untuk_Seleksi_Calon_Mahasiswa_pada_Penerimaan_Mahasiswa_Baru_di_Universitas_Pamulang.

[50] L. Nilawati and Martin, “Penilaian Apartemen Pada Perusahaan Konsultan Properti Menggunakan Metode Naïve Bayes,” *Inf. Syst. Educ. Prof.*, vol. 4, no. 2, pp. 114–123, 2020.

LAMPIRAN

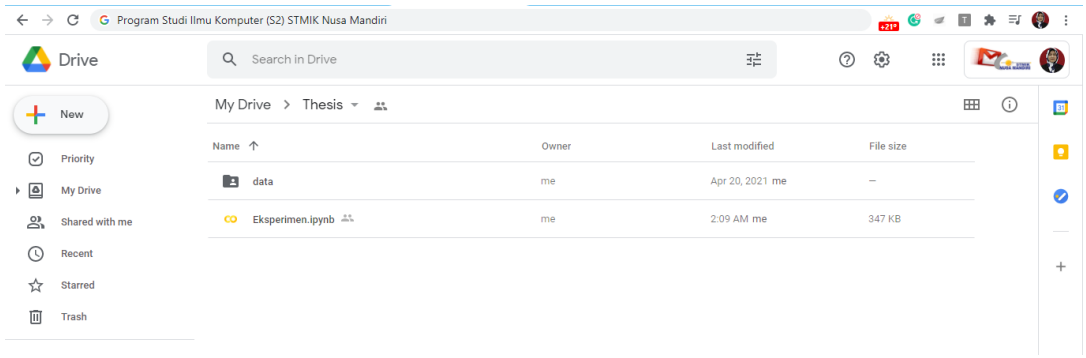
Lampiran 1. Sample Dataset Mobil Bekas 10 Atribut pertama

No	dateCrawled	name	seller	offerType	abtest	vehicleType	yearOfRegistration	gearbox	powerPS	model
1	24/03/2016 11:52	Golf_3_1.6	privat	Angebot	test		1993	manuell	0	golf
2	24/03/2016 10:58	A5_Sportback_2.7_Tdi	privat	Angebot	test	coupe	2011	manuell	190	
3	14/03/2016 12:52	Jeep_Grand_Cherokee_"Overland"	privat	Angebot	test	suv	2004	automatik	163	grand
4	17/03/2016 16:54	GOLF_4_1_4__3TÜRER	privat	Angebot	test	kleinwagen	2001	manuell	75	golf
5	31/03/2016 17:25	Skoda_Fabia_1.4_TDI_PD_Classic	privat	Angebot	test	kleinwagen	2008	manuell	69	fabia
6	04/04/2016 17:36	BMW_316i___e36_Limousine___Bastlerfahrzeug__Export	privat	Angebot	test	limousine	1995	manuell	102	3er
7	01/04/2016 20:48	Peugeot_206_CC_110_Platinum	privat	Angebot	test	cabrio	2004	manuell	109	2_reihe
8	21/03/2016 18:54	VW_Derby_Bj_80__Scheunenfund	privat	Angebot	test	limousine	1980	manuell	50	andere
9	04/04/2016 23:42	Ford_C___Max_Titanium_1_0_L_EcoBoost	privat	Angebot	control	bus	2014	manuell	125	c_max
10	17/03/2016 10:53	VW_Golf_4_5_tuerig_zu_verkaufen_mit_Anhaengerkupplung	privat	Angebot	test	kleinwagen	1998	manuell	101	golf
11	26/03/2016 19:54	Mazda_3_1.6_Sport	privat	Angebot	control	limousine	2004	manuell	105	3_reihe
12	07/04/2016	Volkswagen_Passat	privat	Angebot	control	kombi	2005	manuell	140	passat

Lampiran 3. Link Repositori Koding dan Dataset

Link:

https://drive.google.com/drive/folders/1EyJ62xKOQxfQNSRw-GGt8_M6MbqV3w0_?usp=sharing



Lampiran 4. Sampel hasil prediksi harga hasil transformasi

	Actual_price	predict_price	Diff
105384	9.740380	9.789066	-0.048686
89046	9.099409	9.166513	-0.067105
31295	7.863267	7.487940	0.375327
93947	8.716044	8.415888	0.300156
38730	8.006368	7.891790	0.114578
353932	8.839277	9.023153	-0.183877
368546	9.769956	9.277290	0.492666
39883	7.003065	6.929870	0.073195
179599	8.411833	8.059879	0.351953
112083	7.803843	7.794296	0.009547

Lampiran 5. Sampel hasil prediksi harga asli

	Actual_price	predict_price	Diff
105384	16990.0	17837.642578	-847.642578
89046	8950.0	9571.196289	-621.196289
31295	2600.0	1786.368042	813.631958
93947	6100.0	4518.285156	1581.714844
38730	3000.0	2675.228027	324.771973
353932	6900.0	8292.885742	-1392.885742
368546	17500.0	10692.419922	6807.580078
39883	1100.0	1022.361206	77.638794
179599	4500.0	3164.907959	1335.092041
112083	2450.0	2426.720947	23.279053