

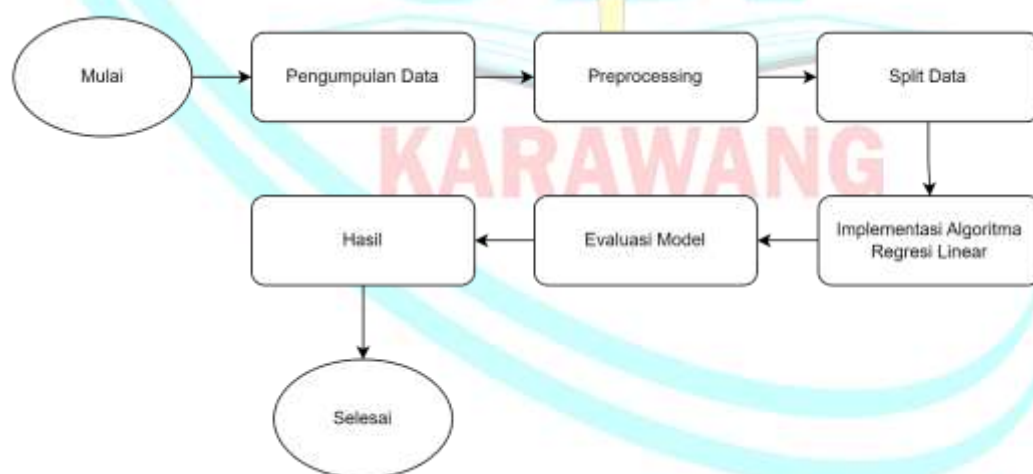
BAB III METODE PENELITIAN

3.1 Objek Penelitian

Objek Penelitian ini berfokus pada prediksi penjualan sepeda motor, yang dibagi menjadi tujuh kategori Bulan, Tahun, Tipe Motor, Nama Motor, Pembayaran, Jumlah Unit Terjual dan Harga Motor. Penelitian ini bertujuan untuk mengembangkan metode prediksi dalam menentukan tingkat akurasi dan peramalan penjualan sepeda motor di Star Motor III Telagasari. Data yang digunakan pada penelitian ini dari Star Motor III Telagasari.

3.2 Prosedur Penelitian

Prosedur penelitian ini disusun untuk memberikan langkah-langkah yang terstruktur dalam proses pengumpulan dan analisis data, sehingga tujuan penelitian dapat tercapai. Berikut prosedur penelitian yang dilakukan digambarkan dalam Gambar 3.1.



Gambar 3.1 Prosedur Penelitian

3.2.1 Pengumpulan Data

Penelitian ini menggunakan data primer yang diperoleh langsung dari laporan penjualan sepeda motor Star Motor III Telagasari selama periode 2020 hingga 2024. Dataset terdiri dari 2.257 entri dengan atribut seperti bulan, tahun, tipe motor, nama motor, metode pembayaran, dan jumlah unit yang terjual. Data tersebut kemudian diolah menggunakan *Google Collaboratory* untuk memprediksi

penjualan sepeda motor berdasarkan pola historis. Atribut independen yang digunakan meliputi bulan, tahun, tipe motor, nama motor, dan metode pembayaran, sedangkan variabel dependen adalah jumlah unit terjual. Berikut adalah laporan data penjualan dari tahun 2020 hingga 2024 di Star Motor III Telagasari.

Tabel 3.2 Data Penjualan Sepeda Motor

N o	Bulan	Tahun	Tipe Motor	Nama Motor	Metode Pembayaran	Jumlah unit terjual	Harga
1.	Januari	2020	A1F02 N36S2 A/T	Vario 125	FIF	12	25,500.000
2.	Januari	2020	A5C02 R37S1 AM/T	CB150R StreetFir e	TUNAI	1	28,440.000
3.	Januari	2020	B5D02 M29M 3M/T	CB150 Verza CW	FIF	5	20,940.000
4.	Januari	2020	C1M0 2N41L 0A/T	Genio CBS	ADIRA	22	20,125.000

3.2.2 Preprocessing

Data akan dipreprocessing untuk memastikan kualitasnya dalam memprediksi penjualan sepeda motor. Langkah-langkah preprocessing yang akan dilakukan meliputi duplikat data, *missing value*, *encoding* data kategorikal, Normalisasi Data Numerik.

1. Duplikat data

Proses ini memisahkan data ke dalam delapan kolom terpisah untuk mempermudah pengolahan dan analisis. Kolom-kolom tersebut meliputi Nomor urut, Bulan, Tahun, Tipe motor, Pembayaran, Jumlah unit terjual, Harga. Selanjutnya memeriksa duplikasi pada dataset menggunakan fungsi `.duplicated()` dari *library* pandas. Fungsi ini memastikan tidak ada baris yang identik pada semua kolom. Jika ditemukan duplikasi, berikut langkah yang dapat dilakukan:

Case Duplikat Data

Mulai

#Langkah 1: Ambil dataset mentah Pisahkan menjadi delapan kolom

#Langkah 2: Pemeriksaan Duplikasi Periksa dataset menggunakan fungsi
.duplicated()

#Langkah 3: Penanganan Duplikasi Jika ditemukan duplikasi hapus dengan
.drop_duplicates() ATAU gabungkan data dengan menjumlahkan/rata-rata
ATAU tambahkan kolom baru sebagai penanda

#Langkah 4: Verifikasi Data Periksa Kembali dataset untuk memastikan
data

Selesai

Hasil:

Tabel 3.3 Data Sebelum Duplikasi

N o	Bulan	Tahun	Tipe Motor	Nama Motor	Metode Pembayaran	Jumlah unit terjual	Harga
1	Januari	2020	A1F02N 36S2A/ T	Vario 125	FIF	12	25,500.000
2	Januari	2020	A5C02 R37S1A M/T	CB150R StreetFire	TUNAI	1	28,440.000
2	Januari	2020	A5C02 R37S1A M/T	CB150R StreetFire	TUNAI	1	28,440.000

Tabel 3.4 Data Setelah Penanganan Duplikasi

N o	Bulan	Tahun	Tipe Motor	Nama Motor	Metode Pembayara n	Jumlah unit terjual	Harga
1	Januari	2020	A1F02N 36S2A/ T	Vario 125	FIF	12	25,500.000
2	Januari	2020	A5C02 R37S1A M/T	CB150R StreetFire	TUNAI	1	28,440.000

2. Missing Value

Missing value dapat muncul karena beberapa alasan, seperti kesalahan pencatatan, informasi yang tidak tersedia, atau masalah teknis saat pengumpulan data. Pada dataset ini terdiri dari 1.049 entri dengan satu kolom yang berisi berbagai informasi, seperti bulan, tahun, tipe motor, jumlah unit terjual, dan harga, yang dipisahkan oleh karakter tertentu (delimiter ;). Setelah proses pemisahan, langkah berikutnya adalah memeriksa setiap kolom untuk mengetahui apakah terdapat nilai yang kosong. Jika ada, langkah penanganan bisa dilakukan dengan menghapus baris atau kolom yang terdampak, atau mengisi nilai yang hilang menggunakan metode seperti rata-rata, median, modus, atau nilai tetap tertentu, sesuai konteks data. Berikut adalah proses dalam menangani *missing value*.

Case Missing Value

Mulai

#Langkah 1: Periksa dataset untuk menemukan nilai yang hilang

#Langkah 2: Ganti nilai yang hilang dengan rata-rata kolom terkait (MEAN)
ATAU mode lain sesuai kebutuhan

#Langkah 3: Periksa ulang dataset untuk memastikan tidak ada lagi missing value

Selesai

Hasil:

Tabel 3.5 Data Sebelum Penanganan *Missing Value*

N o	Bulan	Tahun	Tipe Motor	Nama Motor	Metode Pembayaran	Jumlah unit terjual	Harga
1	Januari	2020	A1F02N 36S2A/ T	Vario 125	FIF	12	25,500.000
2	Januari	2020	A5C02 R37S1A M/T	CB150R StreetFire	TUNAI	Nan	28,440.000

Tabel 3.6 Data Setelah Pengananganan *Missing Value*

N o	Bulan	Tahun	Tipe Motor	Nama Motor	Metode Pembayaran	Jumlah unit terjual	Harga
1	Januari	2020	A1F02N 36S2A/ T	Vario 125	FIF	12	25,500.000
2	Januari	2020	A5C02 R37S1A M/T	CB150R StreetFire	TUNAI	5	28,440.000

3. Transformasi Data

Proses encoding pada data kategorikal dalam dataset penjualan sepeda motor dilakukan untuk mengubah data non-numerik menjadi format numerik sehingga dapat digunakan dalam analisis dan pemodelan. Berikut proses encoding data kategorikal:

Case Encoding

Mulai

#Langkah 1: Temukan kolom yang berisikan data kategorikal dalam dataset

#Langkah 2: Pilih metode encoding yang sesuai Label encoding dan One-Hot Encoding

#Langkah 3: Ubah data kategorikal menjadi format numerik menggunakan metode yang dipilih

#Langkah 4: Periksa dataset untuk memastikan kolom kategorikal telah diencoding

Selesai

Hasil:

Tabel 3.7 Hasil Case Encoding Data

Pembayaran	Case Encoding Data
FIF	0

MANDIRI	1
TUNAI	2
SOF	3

4. Normalisasi Data Numerik

Proses normalisasi data numerik pada dataset penjualan sepeda motor bertujuan untuk menyelaraskan skala nilai dari berbagai fitur numerik sehingga berada dalam rentang tertentu, biasanya 0 hingga 1. Langkah ini penting untuk memastikan bahwa perbedaan skala antar fitur tidak memengaruhi hasil analisis atau pemodelan, terutama saat menggunakan algoritma seperti regresi linear, yang sensitif terhadap perbedaan skala dalam data. Dalam dataset ini, kolom seperti jumlah unit terjual dan harga motor merupakan contoh data numerik yang perlu dinormalisasi. Proses ini diawali dengan identifikasi kolom numerik, dilanjutkan dengan penerapan teknik normalisasi, seperti min-max scaling, yang menghitung nilai baru berdasarkan jarak dari nilai minimum dan maksimum dalam kolom tersebut. Dengan normalisasi, data menjadi lebih seimbang dan memungkinkan model regresi linear bekerja lebih optimal dalam menghasilkan prediksi yang akurat. Berikut adalah proses Normalisasi data:

Case Normalisasi Data

Mulai

#Langkah 1: Pilih kolom yang berisikan data numerik dalam dataset

#Langkah 2: Gunakan metode yang sesuai seperti Min-Max Scaling atau Z-Score scaling

#Langkah 3: Ubah nilai setiap kolom numerik menjadi skala baru menggunakan rumus normalisasi

#Langkah 4: Pastikan kolom numerik berada dalam skala yang diinginkan

Selesai

Hasil:

Tabel 3.8 Hasil Normalisasi Data

N o	Bulan	Tahun	Tipe Motor	Nama Motor	Metode Pembayaran	Jumlah unit terjual	Harga
1	Januari	2020	A1F02N 36S2A/ T	Vario 125	FIF	0.006739	-0.248718
2	Januari	2020	A5C02 R37S1A M/T	CB150R StreetFire	TUNAI	-0.332171	-0.248718
3	Januari	2020	A1F02 N36S3 A/T	Vario 125 CBS	SOF	-0.429003	-0.248718

3.3 Split Data

Setelah tahap preprocessing selesai, data dibagi menjadi dua bagian utama: 80% untuk data pelatihan (training) dan 20% untuk data pengujian (testing). Pembagian 80% untuk pelatihan memberikan model cukup data untuk mempelajari pola, hubungan antar fitur, dan variasi dalam dataset, sehingga dapat meningkatkan akurasi prediksi. Sementara itu, 20% data pengujian digunakan untuk mengevaluasi kinerja model pada data yang belum pernah dilihat sebelumnya. Pendekatan ini membantu mengurangi risiko overfitting, di mana model terlalu terfokus pada data pelatihan sehingga sulit beradaptasi dengan data baru. Dengan demikian, penelitian ini menggunakan 80% data latih dan 20% data uji untuk pengujian. Berikut Pseudocode mencari data latih dan uji.

Case Split Data

Mulai

#Langkah 1: Membaca dataset

- Buka file dataset penjualan sepeda motor
- baca file dataset dan simpan dalam variabel 'dataset'

#Langkah 2: Membagi dataset menjadi data latih dan data latih

- Tentukan proporsi data latih 0.8 dan data uji 0.2
- Hitung jumlah total sampel dalam dataset
- Hitung jumlah sampel untuk data latih
- Hitung jumlah sampel untuk data uji
- Acak urutan dataset untuk memastikan data tidak berurutan
- Ambil sejumlah sampel pertama sebagai data latih
- Ambil sisa sampel sebagai data uji

#Langkah 3: Mengembalikan data latih dan data uji

- Kembalikan data latih
- Kembalikan data uji

Selesai

Hasil:

Tabel 3.9 Hasil Data Latih

No	Bulan	Tahun	Tipe Motor	Nama Motor	Metode Pembayaran	Jumlah unit terjual	Harga
1	Januari	2020	A1F02N36S2A/T	Vario 125	FIF	12	23675000
3	Januari	2020	A5C02R37S1A M/T	CB150R StreetFire	TUNAI	3	23675000
4	Januari	2020	A1F02N36S3A/T	Vario 125 CBS	SOF	8	23675000
6	Januari	2020	A1F02N36S3A/T	Vario 125 CBS	MANDIRI	1	23675000
7	Januari	2020	A1F02N36S3A/T	Vario 125 CBS	SOF	2	23675000

Tabel 3.10 Hasil Data Uji

No	Bulan	Tahun	Tipe Motor	Nama Motor	Metode Pembayaran	Jumlah unit terjual	Harga
----	-------	-------	------------	------------	-------------------	---------------------	-------

2	Januari	2020	A1F02N 36S2A/ T	Vario 125 CBS	MANDIRI	5	23675000
5	Januari	2020	A5C02 R37S1A M/T	CB150R StreetFire	FIF	2	23675000
14	Februari	2020	A1F02 N37S3 A/T	Vario 125 CBS ISS	FIF	3	25500000
15	Februari	2020	A1F02 N37S3 A/T	Vario 125 CBS ISS	MANDIRI	8	25500000
21	Februari	2020	A1F02 N37S3 A/T	Vario 125 CBS ISS	SOF	1	25500000

3.4 Implementasi Algoritma Regresi Linear

Setelah data selesai melalui proses preprocessing dan dibagi menjadi data *training* serta data *testing*, langkah selanjutnya adalah menggunakan algoritma untuk membangun model prediksi penjualan sepeda motor. Penelitian ini menerapkan algoritma *Regresi Linear* untuk memperkirakan jumlah unit yang terjual dengan memanfaatkan hubungan antara variabel independen dan variabel dependen.

3.4.1 Mempersiapkan Data untuk Model

Langkah awal dalam mempersiapkan data untuk model prediksi adalah menentukan variabel input dan target. Variabel input, yang juga disebut fitur atau variabel independen, mencakup kolom-kolom yang diyakini berpengaruh terhadap hasil prediksi. Pada kasus ini, kolom seperti Bulan, Tahun, Tipe Motor, Nama Motor, Pembayaran, dan Harga dipilih sebagai variabel input karena dianggap memiliki pengaruh terhadap penjualan. Sebaliknya, variabel target, yang dikenal sebagai variabel dependen, adalah kolom Jumlah Unit Terjual, yang menjadi fokus utama dalam prediksi. Model kemudian akan mempelajari hubungan antara variabel input dan target untuk menghasilkan estimasi yang akurat.

3.4.2 Pelatihan Model

Proses pelatihan model dilakukan dengan menggunakan algoritma *Regresi Linear* pada data training yang telah dipersiapkan. Algoritma ini dirancang untuk mengenali pola dan hubungan antara fitur-fitur input, seperti Bulan, Tahun, Type Motor, Nama Motor, Pembayaran, dan Harga, terhadap variabel target, yaitu Jumlah Unit Terjual. Melalui pendekatan linier, algoritma akan menentukan garis atau persamaan terbaik yang dapat merepresentasikan hubungan antara variabel-variabel tersebut. Hasil dari proses ini adalah sebuah model yang mampu melakukan prediksi berdasarkan pola yang telah dipelajari selama pelatihan. Berikut alur penerapan Algoritma pada Prediksi Penjualan Sepeda Motor:

Algoritma 1. Membuat Model dengan Algoritma Regresi Linear

Prediksi *Regresi Linear*

Input : Data Tabel Penjualan Sepeda Motor

Output: Prediksi Penjualan Sepeda Motor

1. Input data latih untuk melatih model regresi linear
 2. Identifikasi variabel independent sebagai input
 3. Gunakan jumlah unit terjual sebagai variabel dependen
 4. Membuat model yang sudah dilatih untuk memprediksi penjualan menggunakan data uji
 5. Hasil Prediksi
-

3.5 Pengujian

Setelah model selesai dilatih menggunakan algoritma Regresi Linear pada data training, langkah selanjutnya adalah menguji model tersebut menggunakan data testing. Tujuan dari pengujian ini adalah untuk menilai sejauh mana model mampu memprediksi jumlah unit terjual pada data baru yang belum pernah digunakan selama pelatihan. Hal ini penting untuk memastikan bahwa model tidak hanya menunjukkan performa baik pada data training, tetapi juga dapat mengaplikasikan pola yang telah dipelajari ke data yang belum dikenal.

Tabel 3.11 Pengujian Data Testing

<i>Pengujian Model Data Testing</i>	
Input	: Data testing
Output:	- Prediksi Penjualan Sepeda Motor
1. Input data testing yang telah dipisahkan sebelumnya digunakan sebagai masukan untuk model 2. Data dilakukan normalisasi dan encoding 3. Gunakan model regresi linear yang telah dilatih untuk memprediksi jumlah unit terjual 4. Hasil prediksi menggunakan matrix seperti MSE, MAE, MAPE dan RMSE	

3.5 Evaluasi Model

Setelah proses pelatihan selesai, langkah selanjutnya adalah menguji kinerja model dengan menggunakan data testing. Pengujian ini dilakukan untuk menilai kemampuan model dalam menghasilkan prediksi yang akurat berdasarkan pola yang telah dipelajari sebelumnya. Untuk mengevaluasi performa model, digunakan beberapa metrik, di antaranya:

3.5.1 Mean Absolute Error (MAE)

Mean Absolute Error (MAE) merupakan salah satu metrik evaluasi dalam machine learning yang digunakan untuk menilai sejauh mana prediksi model mendekati nilai aktual. Metrik ini menghitung rata-rata dari nilai absolut selisih antara prediksi dengan nilai sebenarnya, tanpa memperhatikan arah kesalahan (baik positif maupun negatif). Rumus untuk MAE adalah sebagai berikut:

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

Dimana:

n adalah jumlah data

y_i adalah nilai actual

$\hat{y}_i$ adalah nilai prediksi

$\sum$ menunjukkan penjumlahan dari semua selisih absolut

$\frac{1}{n}$ adalah kebalikan dari jumlah total observasi, yang merupakan rata-rata dari selisih absolut

MAE mengukur rata-rata kesalahan prediksi dalam satuan yang sama dengan variabel yang diamati. Nilai MAE yang lebih rendah menunjukkan bahwa model memiliki kemampuan prediksi yang lebih baik.

3.5.2 Mean Squared Error (MSE)

Root Mean Squared Error (RMSE) adalah metrik evaluasi yang digunakan untuk menilai sejauh mana model mampu membuat prediksi dengan akurasi tinggi. RMSE dihitung dengan mengambil akar kuadrat dari nilai *Mean Squared Error* (MSE). Rumus untuk menghitung RMSE adalah sebagai berikut:

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

Dimana:

MSE rata-rata kuadrat nilai prediksi dan nilai aktual

n adalah jumlah data

y_i adalah nilai actual

$\hat{y}_i$ adalah nilai prediksi

$\sum$ menunjukkan penjumlahan dari semua selisih absolut

$\frac{1}{n}$ adalah kebalikan dari jumlah total observasi, yang merupakan rata-rata dari selisih absolut

3.5.3 Root Mean Squared Error

Root Mean Squared Error (RMSE) sering digunakan sebagai metrik untuk mengevaluasi kinerja prediksi model. Metrik ini mengukur tingkat keakuratan

prediksi dengan membandingkan nilai hasil prediksi terhadap nilai aktual. Semakin rendah nilai RMSE, semakin baik model dalam mencerminkan pola perubahan nilai aslinya. Sebaliknya, jika nilai RMSE tinggi, maka akurasi prediksi model cenderung kurang baik dan jauh dari nilai sebenarnya. Proses evaluasi akurasi prediksi dilakukan dengan menggunakan nilai RMSE sebagai indikator utama.

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (3)$$

Dimana:

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

3.5.4 Mean Absolute Percentage Error (MAPE)

Mean Absolute Percentage Error (MAPE) merupakan metrik dalam analisis statistik yang digunakan untuk mengevaluasi tingkat akurasi sebuah model prediksi atau peramalan. Metrik ini memberikan informasi mengenai sejauh mana hasil prediksi model mendekati nilai aktual yang sebenarnya.

$$MAPE = \sum_{t=1}^n \left| \frac{y_t - \bar{y}}{\bar{y}_1} \right| \times 100\% \quad (4)$$

Dimana:

- Σ menunjukkan penjumlahan dari semua selisih absolut
- n adalah jumlah total data
- t = 1 penjumlahan dimulai pada periode waktu dan berlanjut hingga periode waktu terakhir dalam kumpulan data
- $\frac{y_t - \bar{y}}{\bar{y}_1}$ persentase kesalahan antara nilai actual dan nilai yang diprediksi pada periode atau data ke-i

Nilai MAPE dapat diartikan ke dalam 4 kategori yaitu:

Tabel 3.12 *Range* Nilai MAPE

Range MAPE	Arti Nilai
------------	------------

< 10%	Kemampuan Model Peramalan Sangat baik
10 - 20%	Kemampuan Model Peramalan Baik
20 – 50%	Kemampuan Model Peramalan Layak
>50%	Kemampuan Model Peramalan Buruk

6.9.5 R-Squared

R-squared adalah metrik dalam regresi yang digunakan untuk mengukur seberapa baik model dapat menjelaskan variasi data actual. Nilai R^2 berkisar antara 0 hingga 1, dengan interpretasi sebagai berikut:

- $R^2 = 1$ menunjukkan bahwa model dapat menjelaskan seluruh variasi data dengan sempurna
- $R^2 = 0$ berarti model tidak mampu menjelaskan variasi data lebih baik dibandingkan rata-rata nilai actual

Berikut Rumus R^2 :

$$R^2 = 1 - \frac{S_{Sres}}{S_{Stot}} \quad (5)$$

Dimana:

S_{Sres} (Residual Sum of Squares) adalah jumlah kuadrat selisih antara nilai actual dan nilai prediksi:

$$S_{Sres} = \sum (y_i - \hat{y}_i)^2 \quad (6)$$

S_{Stot} (Total Sum of Squares) adalah jumlah kuadrat selisih antara nilai actual dengan rata-rata nilai actual

$$S_{Stot} = \sum (y_i - \bar{y})^2$$

Nilai ini menunjukkan total variasi dalam dataset

R^2 juga dapat dihitung dengan menggunakan korelasi antara variabel independen dan dependen:

$$R^2 = \left(\frac{\sum (y_i - \bar{y})(\hat{y}_i - \bar{\hat{y}})}{\sqrt{\sum (y_i - \bar{y})^2 \sum (\hat{y}_i - \bar{\hat{y}})^2}} \right)^2 \quad (7)$$

Untuk model dengan banyak variabel independen, digunakan Adjusted R^2 untuk menghindari bias:

$$R^2_{adj} = 1 - \left(\frac{(1 - R^2) - (n - 1)}{n - k - 1} \right) \quad (8)$$

Dimana:

n jumlah data

k jumlah variabel independen



