

TUGAS BESAR
MATA KULIAH PENAMBANGAN DATA
Dosen Pengampu: Dr. Abdullah Fajar, S.Si., M.Sc.



Disusun Oleh:
SI 47 05 - Kelompok 3

**"Prediksi Harga Produk Ritel dan Pengelompokan Data Barang
Menggunakan Pendekatan Regresi Linear, Decision Tree dan K-Means"**

Egi Agung Santoso Pardede 102022300266

Fabert Varico 102022300432

Firdaus Yudha Sakti 102022300181

Yunky Novredly 102022330283

PROGRAM STUDI SISTEM INFORMASI
FAKULTAS REKAYASA INDUSTRI
UNIVERSITAS TELKOM
BANDUNG
2025

KATA PENGANTAR

Puji syukur kami panjatkan ke hadirat Tuhan Yang Maha Esa atas rahmat dan karunia-Nya, sehingga laporan Tugas Besar mata kuliah Penambangan Data ini dapat kami selesaikan dengan baik. Laporan ini berjudul "**Prediksi Harga Produk Ritel dan Pengelompokan Data Barang Menggunakan Pendekatan K-Means, Regresi Linear, dan Decision Tree**".

Penyusunan laporan ini bertujuan untuk memenuhi salah satu syarat penilaian akhir pada mata kuliah Penambangan Data. Dalam tugas besar ini, kami menerapkan berbagai teknik penambangan data, baik *unsupervised learning* (K-Means) maupun *supervised learning* (Regresi Linear dan Decision Tree), untuk mengidentifikasi pola pengelompokan produk dan melakukan prediksi harga berdasarkan data yang telah dianalisis.

Kami mengucapkan terima kasih yang sebesar-besarnya kepada Bapak **Dr. Abdullah Fajar, S.Si., M.Sc** selaku dosen pengampu mata kuliah Penambangan Data, serta semua pihak yang telah memberikan bimbingan, dukungan, dan masukan berharga, baik secara langsung maupun tidak langsung, selama proses penyusunan laporan ini.

Kami menyadari bahwa laporan ini masih jauh dari sempurna. Oleh karena itu, segala bentuk kritik dan saran yang membangun akan kami terima dengan lapang dada demi penyempurnaan di masa mendatang.

Besar harapan kami, laporan ini dapat memberikan manfaat, wawasan, serta kontribusi positif bagi para pembaca, khususnya dalam memahami aplikasi penambangan data di bidang e-commerce.

Bandung, 01 Juni 2025

DAFTAR ISI

KATA PENGANTAR	i
DAFTAR ISI.....	ii
DAFTAR GAMBAR.....	iv
BAB 1 PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah	2
1.3 Tujuan.....	2
1.4 Manfaat.....	2
1.5 State of The Art	3
1.5.1 Metode Unsupervised : K-Means.....	3
1.5.2 Metode Supervised : Decision Tree & Regresi	4
BAB 2 KERANGKA KERJA.....	5
2.1 Business Understanding	5
2.2 Sumber Data dan Dataset	5
2.3 Data Exploratory Analysis (Exploratory Data Analysis)	6
2.3.1 Memahami Struktur Dataset.....	6
2.3.2 Tipe Data	7
2.3.3 Visualisasi Data.....	7
2.4 Data Preparation	7
2.4.1 Cleaning Data	7
2.4.2 Penanganan Missing Value	8
2.4.3 Outlier.....	10
2.5 Modelling dan Evaluation	13
2.5.1 K-Means Clustering.....	13
2.5.2 Regression Linear.....	17
2.5.3 Decision Tree.....	21
2.6 Mockup dashboard	24
BAB 3 HASIL DAN PEMBAHASAN	25
3.1 Analisis Hasil Evaluasi Model	25
3.1.1 Hasil Model Prediksi	25
3.1.1.1 Linear Regression.....	25
3.1.1.2 Analisis Linear Regression.....	26
3.1.1.3 Decision Tree.....	26
3.1.1.4 Analisis Decision Tree	27
3.1.1.5 Keuntungan dan Keterbatasan Model	28
3.1.2 Hasil Model Clustering.....	29
3.1.2.1 Proses Clustering.....	29

3.1.2.2 Hasil Clustering	29
3.1.2.3 Silhouette Score.....	30
3.1.2.4 Visualisasi Hasil Clustering	30
3.1.2.5 Analisis	31
3.1.2.6 Keuntungan dan Keterbatasan	31
3.1.3 Analisis Penggunaan Model.....	32
3.1.4 Analisis Evaluasi	32
3.2 Dashboard.....	33
3.2.1 Tujuan Dashboard	33
3.2.2 Isi Dashboard.....	34
3.2.2.1 Tampilan Dashboard	34
3.2.2.1.1 K-Means Model Info	34
3.2.2.1.2 K-Means Feature Analysis	34
3.2.2.1.3 K-Means Prediksi Harga	35
3.2.2.1.4 K-Means Visualisasi Model	36
3.2.2.1.5 Analisis Data	37
3.2.2.1.6 Decision Tree Model Info	38
3.2.2.1.7 Decision Tree Feature Analysis.....	38
3.2.2.1.8 Decision Tree Prediksi Harga.....	39
3.2.2.1.9 Decision Tree Visualisasi Model	40
3.2.2.1.10 Linear Regression Model Info.....	41
3.2.2.1.11 Linear Regression Feature Analysis.....	41
3.2.2.1.12 Linear Regression Prediksi Harga.....	42
3.2.2.1.13 Linear Regression Visualisasi Model.....	43
3.2.2.2 Penjelasan Fitur Dashboard	43
BAB 4 PENUTUP.....	45
4.1 Kesimpulan.....	45
4.2 Saran.....	45
DAFTAR PUSTAKA	47
LAMPIRAN	48

DAFTAR GAMBAR

Figure 2. 1 Mengidentifikasi nilai NaN.....	8
Figure 2. 2 Menghitung dan menampilkan jumlah serta persentase missing value pada kolom dataset	8
Figure 2. 3 Jumlah missing value pada tiap kolom dataset	9
Figure 2. 4 Jumlah dan persentase missing value pada tiap kolom dataset	9
Figure 2.5 Mengisi missing value pada kolom dengan modus	10
Figure 2. 6 Menghitung lower boundary dan upper boundary	11
Figure 2. 7 Mendeteksi apakah ada outlier pada kolom sale_price.....	11
Figure 2. 8 Mendeteksi apakah ada outlier pada kolom market_price	11
Figure 2. 9 Outlier pada kolom sale_price	11
Figure 2. 10 Outlier pada kolom market_price.....	11
Figure 2. 11 Memvisualisasi distribusi, normalitas dan sebaran data pada kolom market_price	12
Figure 2. 12 Memvisualisasi distribusi, normalitas dan sebaran data pada kolom sale_price	12
Figure 2. 13 Hasil visualisasi pada kolom market_price.....	13
Figure 2. 14 Hasil visualisasi pada kolom sale_price.....	13
Figure 2. 15 Memilih kolom sale_price dan market_price untuk analisis clustering.	13
Figure 2. 16 Visualisasi data setelah standarisasi menggunakan StandardScaler, dengan setiap fitur memiliki rata-rata 0 dan standar deviasi 1	14
Figure 2. 17 Menyiapkan variabel untuk menyimpan nilai inertia dan silhouette score pada rentang jumlah cluster dari 2 hingga 10	14
Figure 2. 18 Menjalankan K-Means untuk berbagai jumlah klaster, menghitung inertia dan silhouette score untuk evaluasi kualitas klaster	14
Figure 2. 19 Mengelompokkan data berdasarkan klaster dan menghitung rata-rata serta jumlah sale_price dan rata-rata market_price untuk setiap klaster	16
Figure 2. 20 Menghitung rata-rata dan jumlah harga jual serta harga pasar untuk setiap klaster	16
Figure 2. 21 Ringkasan Cluster	17
Figure 2. 22 Menetapkan kolom market_price sebagai variabel independen (X) dan kolom sale_price sebagai variabel dependen (y).....	17
Figure 2. 23 Membagi data menjadi dua bagian: 80% untuk pelatihan (X_train, y_train) dan 20% untuk pengujian (X_test, y_test)	18
Figure 2. 24 Membuat model regresi linear dan melatihnya menggunakan data pelatihan (X_train, y_train)	18
Figure 2. 25 Memprediksi sale_price menggunakan model regresi linear pada data uji X_test	18
Figure 2. 26 Menghitung nilai MSE, RMSE, dan R ² (R-squared) untuk	

mengevaluasi kinerja model regresi	19
Figure 2. 27 Mencetak hasil analisis regresi linear, termasuk intercept (b0), koefisien (b1), nilai R-squared (R^2), dan nilai RMSE untuk mengevaluasi akurasi model	19
Figure 2. 28 Hasil analisis Regresi Linear.....	20
Figure 2. 29 Membagi data menjadi set pelatihan (80%) dan pengujian (20%)	21
Figure 2. 30 Membagi data menjadi dua set: 80% untuk pelatihan (X_train, y_train) dan 20% untuk pengujian (X_test, y_test).....	21
Figure 2. 31 Membuat dan melatih model Decision Tree dengan parameter tertentu untuk memprediksi harga jual	21
Figure 2. 32 Memprediksi harga jual dengan memastikan bahwa harga jual selalu lebih tinggi minimal 10% dari harga pasar	22
Figure 2. 33 Melakukan prediksi harga jual menggunakan fungsi kustom yang memastikan harga jual lebih tinggi minimal 10% dari harga pasar	22
Figure 2. 34 Menghitung evaluasi model menggunakan MSE, RMSE, dan R-squared untuk mengukur akurasi prediksi."	22
Figure 2. 35 Hasil analisis Decision Tree.....	23
Figure 2. 36 Mockup dashboard	24
Figure 3.1 Tampilan Dashboard K-Means Model Info	34
Figure 3.2 Tampilan Dashboard Analisis Feature	34
Figure 3.3 Tampilan Dashboard Analisis Margin	35
Figure 3.4 Tampilan Dashboard Prediksi Harga	35
Figure 3.5 Tampilan Dashboard Hasil Clustering KMeans	36
Figure 3.6 Tampilan Dashboard Elbow Method	36
Figure 3.7 Tampilan Dashboard Silhouette Score.....	37
Figure 3.8 Tampilan Dashboard Business Understanding	37
Figure 3.10 Tampilan Dashboard Analisis Feature Decision Tree	38
Figure 3.12 Tampilan Dashboard Prediksi Harga Decision Tree.....	40
Figure 3.13 Tampilan Dashboard Pohon Keputusan Decision Tree	40
Figure 3.14 Tampilan Dashboard Prediksi vs Aktual Decision Tree.....	41
Figure 3.16 Tampilan Dashboard Analisis Feature	41
Figure 3.18 Tampilan Dashboard Prediksi Harga Linear Regression	43
Figure 3.19 Tampilan Dashboard Koefisien Linear Regression	43
Figure 3.20 Memilih file model (.pkl).....	43
Figure 3.21 Menampilkan hasil prediksi harga Linear Regression	44
Figure 3.22 Menampilkan hasil prediksi harga Decision Tree.....	44
Figure 3.23 Menampilkan hasil Feature Importance	44

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Dalam era digital yang terus berkembang, bisnis e-commerce atau toko online telah menjadi bagian integral dari perekonomian modern. Pertumbuhan pesat platform penjualan daring menciptakan tantangan sekaligus peluang baru bagi para pelaku bisnis. Untuk tetap kompetitif dan berkelanjutan, penting bagi toko online untuk tidak hanya menawarkan produk yang menarik, tetapi juga memahami secara mendalam bagaimana produk-produk tersebut berkinerja di pasar digital.

Sebuah toko online yang menyediakan berbagai produk kebutuhan rumah menghadapi volume data transaksi dan produk yang signifikan. Data ini, yang mencakup informasi seperti nama produk, kategori, harga jual (sale price), dan harga pasar (market price), merupakan aset berharga yang dapat diolah untuk menghasilkan wawasan bisnis yang strategis. Namun, mengelola dan menafsirkan data mentah ini secara manual adalah tugas yang kompleks dan memakan waktu.

Tantangan utama yang dihadapi adalah bagaimana mengidentifikasi produk-produk unggulan, memahami dinamika penentuan harga (terutama perbandingan antara harga jual dan harga pasar), mengidentifikasi anomali atau outlier dalam data, serta mengelompokkan produk berdasarkan karakteristik harga untuk strategi pemasaran atau penentuan harga yang lebih efektif. Selain itu, memprediksi harga jual yang optimal berdasarkan harga pasar menjadi krusial untuk memaksimalkan keuntungan dan daya saing. Tanpa analisis data yang tepat, keputusan bisnis terkait inventaris, promosi, dan penentuan harga cenderung bersifat spekulatif dan kurang optimal.

Oleh karena itu, diperlukan pendekatan sistematis menggunakan teknik analisis data dan pemodelan statistik untuk menggali informasi tersembunyi dari data produk. Analisis ini bertujuan untuk memberikan pemahaman yang jelas tentang performa setiap produk dan kategori, serta menghasilkan rekomendasi berbasis data yang dapat mendorong pertumbuhan bisnis toko online.

1.2 Rumusan Masalah

Berdasarkan latar belakang di atas, rumusan masalah dalam proyek analisis data produk ini adalah sebagai berikut:

1. Bagaimana menganalisis dan memvisualisasikan distribusi serta karakteristik statistik dari harga jual (`sale_price`) dan harga pasar (`market_price`) pada data produk?
2. Bagaimana mengidentifikasi dan menangani nilai yang hilang (`missing values`) dan nilai ekstrim (`outliers`) dalam dataset produk?
3. Bagaimana menganalisis jumlah dan distribusi produk berdasarkan kategori, sub-kategori, merek, dan tipe?
4. Bagaimana membandingkan harga jual rata-rata (`sale_price`) dengan harga pasar rata-rata (`market_price`), serta menghitung potensi keuntungan (`profit`) atau margin dari setiap produk atau kategori?
5. Bagaimana mengelompokkan produk (`clustering`) berdasarkan karakteristik harga (`sale_price` dan `market_price`) menggunakan algoritma K-Means untuk mengidentifikasi segmen produk dengan pola harga serupa?
6. Bagaimana membangun model regresi linear untuk memprediksi harga jual (`sale_price`) berdasarkan harga pasar (`market_price`)?
7. Bagaimana membangun model Decision Tree untuk memprediksi harga jual (`sale_price`), dengan pertimbangan untuk memastikan harga jual lebih tinggi dari harga pasar?
8. Bagaimana mengevaluasi performa model regresi linear dan Decision Tree menggunakan metrik yang sesuai?
9. Bagaimana hasil analisis dan model yang dibangun dapat memberikan wawasan untuk strategi penentuan harga, manajemen inventaris, atau pengembangan produk di toko online? `market_price`

1.3 Tujuan

Tujuan dari penelitian ini adalah sebagai berikut :

1. Melakukan eksplorasi data mendalam (EDA) untuk memahami struktur, distribusi, dan karakteristik statistik dari data produk.
2. Mengidentifikasi dan menangani masalah kualitas data seperti nilai hilang dan outlier pada dataset.
3. Menganalisis performa produk berdasarkan harga jual, harga pasar, profit/margin, serta distribusi produk per kategori dan sub-kategori.
4. Mengidentifikasi segmen produk dengan pola harga serupa melalui implementasi K-Means clustering.
5. Mengembangkan model prediksi harga menggunakan Regresi Linear dan Decision Tree untuk memprediksi harga jual berdasarkan harga pasar.

6. Mengevaluasi dan membandingkan performa model prediksi yang dibangun.
7. Menyediakan wawasan berbasis data yang dapat mendukung pengambilan keputusan strategis terkait penentuan harga, manajemen inventaris, dan strategi pemasaran bagi toko online.

1.4 Manfaat

Manfaat dari penelitian ini adalah sebagai berikut :

1. Bagi E-commers atau Toko Online:
 1. Memberikan pemahaman yang jelas tentang performa setiap produk dan kategori.
 2. Memberikan dasar data untuk menginformasikan strategi penentuan harga yang lebih efektif dan menguntungkan.
 3. Membantu dalam identifikasi produk berkinerja tinggi atau rendah.
 4. Memberikan wawasan untuk optimasi manajemen inventaris berdasarkan popularitas kategori.
 5. Memberikan dasar untuk pengembangan produk baru atau strategi pemasaran yang ditargetkan berdasarkan segmen produk.
 6. Meningkatkan efisiensi operasional melalui keputusan berbasis data.

1.5 State of The Art

Kami menggunakan kombinasi dari pendekatan unsupervised dan supervised untuk mendapatkan hasil analisis yang optimal. Pemilihan algoritma ini didasarkan pada karakteristik data yang dianalisis serta tujuan akhir dari penelitian.

1.5.1 Metode Unsupervised : K-Means

Pada tahap analisis unsupervised learning, kami menggunakan algoritma K-Means Clustering. Algoritma ini dipilih karena kemampuannya dalam mengelompokkan data ke dalam beberapa cluster berdasarkan kesamaan fitur numerik. Dalam konteks data produk toko online ini, K-Means diterapkan untuk mengidentifikasi kelompok produk yang memiliki pola harga jual (*sale_price*) dan harga pasar (*market_price*) yang serupa. [1] Pengelompokan ini bertujuan untuk memberikan wawasan mengenai struktur data harga produk, yang nantinya dapat membantu dalam strategi penentuan harga atau identifikasi segmen produk.

K-Means bekerja dengan cara meminimalkan jarak antara setiap titik data dengan pusat (*centroid*) cluster tempatnya berada, sehingga data dengan karakteristik harga yang mirip terkumpul dalam satu cluster. [1]

Langkah-langkah utama yang kami lakukan dalam implementasi K-Means di sini meliputi:

1. Persiapan Data: Memilih fitur numerik yang relevan untuk clustering, yaitu *sale_price* dan *market_price*. Data dari kedua fitur ini kemudian distandarisasi menggunakan *StandardScaler* agar memiliki skala yang serupa, karena K-Means sensitif terhadap skala fitur.
2. Menentukan Jumlah Cluster (*k*) Optimal: Menggunakan metode *Elbow Method* dan *Silhouette Score* untuk mengevaluasi performa clustering pada rentang jumlah cluster (*k*) yang berbeda (dalam hal ini dari *k*=2 hingga *k*=10). Visualisasi inertia (*sum of squared distances*) dan nilai *silhouette score* membantu menentukan jumlah cluster yang paling sesuai dengan struktur data.

3. Penerapan K-Means: Setelah menentukan jumlah cluster (berdasarkan analisis pada langkah 2), algoritma K-Means dijalankan pada data yang telah distandarisasi dengan jumlah cluster yang dipilih (misalnya, $k=3$ berdasarkan hasil plot).
4. Penetapan Label Cluster: Hasil clustering (label cluster untuk setiap produk) ditambahkan sebagai kolom baru pada DataFrame asli (df).
5. Analisis Cluster: Melakukan analisis ringkasan (misalnya, nilai rata-rata sale_price dan market_price, serta jumlah produk) untuk setiap cluster untuk memahami karakteristik masing-masing kelompok produk yang terbentuk.

1.5.2 Metode Supervised : Decision Tree & Regresi

Pada tahap analisis supervised learning, kami mengaplikasikan dua algoritma regresi, yaitu Linear Regression dan Decision Tree Regressor, untuk memprediksi harga jual (sale_price) berdasarkan harga pasar (market_price). [1] Kedua model ini dipilih untuk mengeksplorasi hubungan antara harga pasar dan harga jual, serta untuk membangun alat prediksi yang dapat digunakan dalam strategi penentuan harga.

1. **Linear Regression** Algoritma ini digunakan untuk memodelkan hubungan linear antara variabel independen (market_price) dan variabel dependen (sale_price). Model Linear Regression berasumsi bahwa ada hubungan garis lurus antara harga pasar dan harga jual. Dengan metode ini, kami dapat:

- 1) Mengukur kekuatan dan arah hubungan linear antara harga pasar dan harga jual.
- 2) Membangun persamaan regresi sederhana ($\text{sale_price} = b_0 + b_1 * \text{market_price}$) untuk memprediksi harga jual.
- 3) Menginterpretasikan koefisien regresi (b_1) untuk memahami seberapa besar perubahan harga jual dipengaruhi oleh perubahan harga pasar. Penerapan Linear Regression di sini bertujuan untuk mendapatkan model dasar yang menjelaskan hubungan harga secara sederhana dan linear.

2. **Decision Tree** Algoritma ini, yang merupakan jenis Decision Tree untuk tugas regresi, digunakan karena kemampuannya menangkap hubungan non-linear dan sifatnya yang interpretable (meskipun untuk pohon yang besar interpretasinya bisa kompleks). Decision Tree Regressor membangun model berupa struktur pohon yang membagi data berdasarkan nilai fitur (market_price) untuk memprediksi nilai target (sale_price). [1] Dalam implementasinya, model ini dilatih dengan penyesuaian agar prediksi harga jual cenderung lebih tinggi dari harga pasar, sesuai dengan tujuan bisnis untuk mencari keuntungan. [1] Dengan metode ini, kami dapat:

- 1) Mengidentifikasi titik-titik pemisah (split points) pada harga pasar yang berpengaruh signifikan terhadap harga jual.
- 2) Membuat aturan prediksi yang dapat dipahami (berupa jalur pada pohon).
- 3) Membangun model prediksi harga jual yang, dengan modifikasi, dapat mendukung strategi penetapan harga di atas harga pasar. [1]
Penerapan Decision Tree Regressor di sini melengkapi Linear Regression dengan menawarkan pendekatan yang berbeda dan lebih fleksibel dalam memodelkan hubungan harga, serta memungkinkan integrasi aturan bisnis ($\text{harga jual} > \text{harga pasar}$).
tahap

BAB 2

KERANGKA KERJA

2.1 Business Understanding

Proyek analisis ini berlatar belakang sebuah toko online yang bergerak dalam penyediaan berbagai produk kebutuhan rumah. Toko ini memiliki ketersediaan data produk yang kaya, mencakup detail-detail penting seperti nama produk, kategori utama, sub-kategori, merek, harga jual (*sale_price*) yang diterapkan di platform toko, dan estimasi harga produk di pasar umum (*market_price*).

Tujuan utama dari analisis ini adalah untuk mendapatkan pemahaman yang mendalam dan komprehensif mengenai performa produk dari perspektif harga. Berdasarkan ketersediaan data (*sale_price* dan *market_price*), tujuan spesifik yang ingin dicapai meliputi:

Analisis Perbandingan Harga: Memahami hubungan antara harga jual (*sale_price*) dan harga pasar (*market_price*). Ini mencakup identifikasi produk-produk yang dijual di atas, di bawah, atau setara dengan harga pasar, serta analisis potensi margin keuntungan atau kerugian berdasarkan selisih harga.

Identifikasi Performa Produk/Kategori: Menemukan produk atau kategori produk mana yang memiliki harga jual rata-rata tertinggi, yang dapat mengindikasikan popularitas atau nilai premium di mata konsumen, atau kategori mana yang paling menguntungkan (berdasarkan perhitungan profit/margin).

Segmentasi Produk: Mengelompokkan produk-produk berdasarkan pola harga jual dan harga pasar menggunakan teknik clustering untuk mengidentifikasi segmen produk dengan karakteristik harga yang serupa. Segmentasi ini dapat membantu dalam merancang strategi penentuan harga atau promosi yang ditargetkan.

Pengembangan Strategi Penentuan Harga: Menggunakan wawasan dari analisis harga dan hasil clustering untuk menginformasikan strategi penentuan harga di masa depan. Ini juga mencakup pengembangan model prediksi (menggunakan regresi) untuk memprediksi harga jual optimal berdasarkan harga

pasar, dengan pertimbangan khusus untuk memastikan keuntungan.

Data yang digunakan dalam analisis ini meliputi kolom-kolom kunci seperti product, category, sub_category, brand, sale_price, market_price, dan type. Melalui eksplorasi data, pembersihan data, dan penerapan teknik clustering serta model regresi (Linear Regression dan Decision Tree), analisis ini diharapkan dapat memberikan wawasan berbasis data yang actionable untuk mendorong pertumbuhan dan profitabilitas toko online merupakan

2.2 Sumber Data dan Dataset

Analisis ini memanfaatkan dataset yang diperoleh dari Kaggle, sebuah repository online terkemuka yang menyediakan kumpulan data untuk riset dan praktik data science. Dataset yang spesifik digunakan adalah "BigBasket Products", yang diunggah oleh kontributor Kaggle untuk memfasilitasi eksplorasi data produk dari platform e-commerce BigBasket

Dataset "BigBasket Products" berisi detail rinci untuk setiap entri produk, termasuk atribut-atribut seperti:

- 1) Index: Sebuah identifier numerik untuk setiap baris data.
- 2) product: Label deskriptif untuk setiap item.
- 3) category: Kategori besar tempat produk diklasifikasikan.
- 4) sub_category: Kategori turunan atau yang lebih spesifik.
- 5) brand: Nama perusahaan atau merek produk.
- 6) sales_price: Harga penjualan aktual di platform.
- 7) market_price: Perkiraan harga standar di pasar.
- 8) type: Klasifikasi lebih lanjut atau varian produk.
- 9) rating: Metrik penilaian berdasarkan masukan pengguna. yang

2.3 Data Exploratory Analysis (Exploratory Data Analysis)

Data exploration adalah langkah awal dalam analisis data yang bertujuan untuk memahami struktur, pola, dan karakteristik dataset secara menyeluruh. Proses ini melibatkan pemeriksaan atribut, identifikasi anomali, serta analisis statistik dasar untuk mendapatkan wawasan awal. Berikut adalah langkah-langkah eksplorasi data yang dilakukan pada dataset BigBasket:

2.3.1 Memahami Struktur Dataset

Komponen utama dataset ini adalah serangkaian atribut yang memberikan detail tentang setiap produk. Langkah pertama dalam analisis adalah pemeriksaan struktur dasar dataset, yaitu menentukan total jumlah produk (baris) dan jenis informasi yang dikumpulkan (kolom).

Beberapa atribut representatif yang akan diperiksa meliputi:

- a) Index: Identifikasi urutan untuk setiap catatan produk.
- b) product: Nama deskriptif produk.
- c) category: Klasifikasi tingkat tinggi untuk produk.
- d) sub_category: Kategori yang lebih rinci di bawah kategori utama.
- e) brand: Merek dagang produk.
- f) sales_price: Biaya produk saat dibeli di platform.
- g) market_price: Harga referensi di pasar.
- h) type: Klasifikasi tambahan untuk produk.
- i) rating: Skor kepuasan pelanggan. terdiri

2.3.2 Tipe Data

Tipe Setiap atribut dalam dataset ini memiliki tipe data spesifik:

- a) Kolom-kolom product, category, sub_category, brand, dan type memiliki tipe data String.
- b) Sementara itu, kolom index, sales_price, market_price, dan rating menggunakan tipe data Integer.data

2.3.3 Visualisasi Data

Pada Pada tahap ini, visualisasi data digunakan secara ekstensif untuk menggali wawasan awal, memahami distribusi variabel, dan mengidentifikasi potensi anomali dalam dataset. Penggunaan grafik membantu dalam mengkomunikasikan karakteristik data yang kompleks secara intuitif.

Visualisasi yang kami terapkan meliputi:

- 1) Histogram: Digunakan untuk menggambarkan distribusi frekuensi dari variabel numerik (sale_price, market_price). Histogram membantu kita melihat bentuk distribusi data (misalnya, normal, skew), nilai-nilai yang paling sering muncul, dan rentang nilai data.
- 2) Q-Q Plot (Quantile-Quantile Plot): Digunakan untuk membandingkan distribusi variabel numerik dengan distribusi teoritis tertentu, biasanya distribusi normal. Plot ini membantu dalam menilai apakah data mengikuti distribusi normal, yang penting untuk beberapa asumsi dalam pemodelan statistik.
- 3) Boxplot: Memberikan ringkasan visual dari distribusi data numerik, termasuk median, kuartil (Q1 dan Q3), rentang interkuartil (IQR), serta identifikasi potensi outlier. Boxplot sangat efektif untuk membandingkan distribusi variabel yang sama di antara kelompok-kelompok yang berbeda (misalnya, harga per kategori).
- 4) Countplot (Bar Plot Frekuensi Kategorikal): Meskipun tidak

disebutkan di daftar Anda, kode Anda menyertakan countplot atau bar plot untuk memvisualisasikan jumlah produk per kategori. Visualisasi ini penting untuk memahami distribusi produk di seluruh kategori diskrit.

- 5) Scatter Plot: Digunakan untuk mengeksplorasi hubungan antara dua variabel numerik, khususnya antara market_price dan sale_price. Scatter plot membantu melihat pola, tren, dan potensi korelasi antara kedua variabel harga ini, serta memvisualisasikan hasil clustering atau model regresi.

2.4 Data Preparation

Data preparation merupakan sebuah proses dalam analisis data untuk memastikan dataset siap digunakan dalam proses pemodelan. Tahapan ini meliputi:

2.4.1 Cleaning Data

Pada Pembersihan data merupakan langkah fundamental pada awal proses Data Preparation. Tahap ini esensial untuk memastikan dataset bebas dari elemen-elemen yang dapat menurunkan kualitas analisis dan prediksi model. Data mentah yang belum diolah seringkali berisi gangguan (noise), inkonsistensi, entri yang kosong (missing values), atau nilai-nilai ekstrim (outliers). Keberadaan masalah-masalah data ini dapat secara signifikan mengurangi akurasi dan keandalan model, serta berpotensi mengarah pada kesimpulan yang tidak tepat. Oleh karena itu, pembersihan data bertujuan untuk meminimalisir dampak negatif tersebut.

2.4.2 Penanganan Missing Value & Duplicated Data

Untuk memastikan kualitas dan akurasi analisis serta model yang akan dikembangkan, sangat penting untuk menangani *missing value* pada tahap persiapan data. *Missing value* dapat menyebabkan bias, menurunkan akurasi model prediksi, dan bahkan menghambat proses analisis. Jika tidak ditangani dengan baik, data yang hilang dapat menyebabkan kesimpulan yang keliru atau model yang tidak dapat diandalkan. Karena itu, langkah-langkah seperti mengidentifikasi, menangani, atau menghapus nilai yang

tidak ada diperlukan sangat penting untuk proses ini.

```
df.isna().sum()
```

```
len(df.drop_duplicates()) / len(df)
```

```
1.0
```

```
duplicates = df[df.duplicated(keep=False)]  
  
print("Baris dengan duplikat:")  
duplicates
```

Baris dengan duplikat:

index	product	category	sub_category	brand	sale_price	market_price	type	rating	description
-------	---------	----------	--------------	-------	------------	--------------	------	--------	-------------

Figure 2. 1 Mengidentifikasi jumlah missing values per kolom

```
total_rows = len(df)  
for column in df.columns:  
    missing_count = df[column].isna().sum()  
    missing_percentage = (missing_count / total_rows) * 100  
    print(f"Column '{column}' Has {missing_count} missing values ({missing_percentage:.2f}%")
```

Figure 2. 2 Menghitung serta persentase missing value pada kolom dataset

Kode tersebut bertujuan untuk mendapatkan gambaran mengenai distribusi missing values di setiap kolom DataFrame (df), menyajikan baik kuantitas maupun proporsinya dalam persentase. Awalnya, agregasi jumlah nilai kosong (NaN) per kolom dilakukan menggunakan `df.isna().sum()`. Selanjutnya, total baris DataFrame diperoleh via `len(df)`. Melalui perulangan pada setiap kolom, jumlah nilai yang hilang (`missing_count`) dihitung ulang menggunakan `df[column].isna().sum()`, setelah itu persentase nilai yang hilang (`missing_percentage`) dikalkulasi dengan ekspresi $(\text{missing_count} / \text{total_rows}) \times 100$. Informasi ini kemudian disajikan secara individual untuk setiap kolom tersebut.

index	0
product	1
category	0
sub_category	0
brand	1
sale_price	0
market_price	0
type	0
rating	8626
description	115

dtype: int64

Figure 2. 3 Jumlah missing value pada tiap kolom dataset

```

Column 'index' Has 0 missing values (0.00%)
Column 'product' Has 1 missing values (0.00%)
Column 'category' Has 0 missing values (0.00%)
Column 'sub_category' Has 0 missing values (0.00%)
Column 'brand' Has 1 missing values (0.00%)
Column 'sale_price' Has 0 missing values (0.00%)
Column 'market_price' Has 0 missing values (0.00%)
Column 'type' Has 0 missing values (0.00%)
Column 'rating' Has 8626 missing values (31.30%)
Column 'description' Has 115 missing values (0.42%)

```

Figure 2. 4 Jumlah dan persentase missing value pada tiap kolom dataset

Berdasarkan output yang didapatkan terlihat bahwa terdapat *missing value* pada dataset untuk mengatasinya digunakan kode berikut.

```

df['product'].fillna(df['product'].mode()[0], inplace=True)
df['brand'].fillna(df['brand'].mode()[0], inplace=True)
df['description'].fillna(df['description'].mode()[0], inplace=True)

```

Figure 2.5 Mengisi missing value pada kolom dengan modus

2.4.3 Outlier

Penanganan outlier, dilakukan setelah pembersihan data dan penanggulangan missing values, memegang peran penting dalam analisis data dan efektivitas model machine learning. Outlier didefinisikan sebagai pengamatan yang nilainya berbeda secara signifikan dari observasi lainnya. Anomali data ini dapat memengaruhi performa model prediktif dengan mengubah metrik statistik deskriptif, seperti rata-rata dan standar deviasi. Lebih lanjut, outlier berisiko mengurangi akurasi model, menyebabkan hasil yang tidak representatif, dan berkontribusi pada fenomena overfitting data dibersihkan dan *missing value* diatasi, penanganan

```
def find_outlier_boundary(df, variable):  
  
    IQR = df[variable].quantile(0.75) - df[variable].quantile(0.25)  
  
    lower_boundary = df[variable].quantile(0.25) - (IQR * 1.5)  
    upper_boundary = df[variable].quantile(0.75) + (IQR * 1.5)  
  
    return upper_boundary, lower_boundary
```

Figure 2. 6 Menghitung lower boundary dan upper boundary

Kode atau skrip diatas merupakan fungsi bernama `find_outlier_boundary` digunakan untuk menghitung batas atas dan bawah dari sebuah variable dalam DataFrame untuk mendeteksi outlier berdasarkan metode IQR.

```
full_occup_upper_limit, full_occup_lower_limit = find_outlier_boundary(df, 'sale_price')
full_occup_upper_limit, full_occup_lower_limit
```

Figure 2. 7 pengecekan apakah ada outlier pada kolom sale_price

```
full_occup_upper_limit, full_occup_lower_limit = find_outlier_boundary(df, 'market_price')
full_occup_upper_limit, full_occup_lower_limit
```

Figure 2. 8 Pengecekan apakah ada outlier pada kolom market_price

Dari kode tersebut digunakan untuk menghitung dan menampilkan batas atas dan bawah dari outlier pada kolom `sale_price` dan `market_price` dalam DataFrame menggunakan fungsi `find_outlier_boundary`. Dari kode diatas didapatkan hasil berikut.

```
(np.float64(755.0), np.float64(-301.0))
```

Figure 2.9 Outlier pada kolom sale_price

```
(np.float64(912.5), np.float64(-387.5))
```

Figure 2. 10 Outlier pada kolom market_price

Berdasarkan perhitungan menggunakan metode IQR, batas outlier untuk `sale_price` adalah 755 sebagai batas atas dan -301 sebagai batas bawah. Untuk `market_price`, batas atasnya adalah 912.5, sedangkan batas bawahnya adalah -387.5.

```
check_plot(data_clf, 'market_price')
```

Figure 2. 11 Memvisualisasi distribusi, normalitas dan sebaran data pada kolom market_price

```
check_plot(data_clf, 'sale_price')
```

Figure 2. 12 Memvisualisasi distribusi, normalitas dan sebaran data pada kolom sale_price

Selanjutnya, kode memanggil fungsi `check_plot`, meneruskan `data_clf` (DataFrame yang telah disaring dari outlier) sebagai argumen data, serta nama kolom yang akan divisualisasikan. Fungsi ini berperan menghasilkan serangkaian grafik Histogram, Q-Q Plot, dan Boxplot guna pemahaman distribusi data pada kolom `market_price` dan `sale_price` setelah proses pemfilteran. Dengan visualisasi ini, kita dapat mengevaluasi apakah data yang telah dibersihkan menunjukkan pola distribusi spesifik, memastikan outlier telah tertangani dengan baik, dan memperoleh gambaran visual yang mendukung analisis data sebelum melangkah ke tahapan berikutnya. Visualisasinya disajikan sebagai berikut

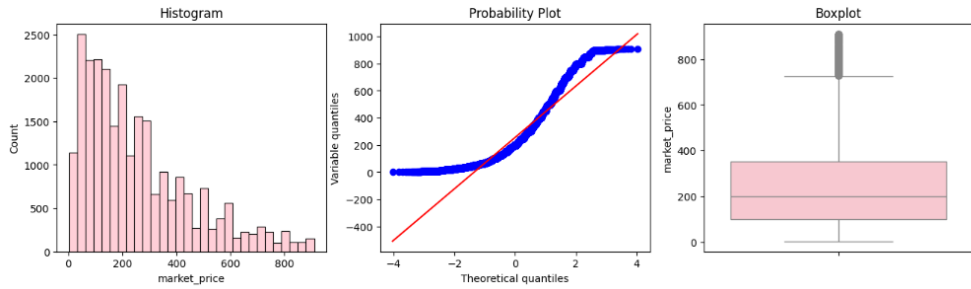


Figure 2. 13 Hasil visualisasi pada kolom market_price

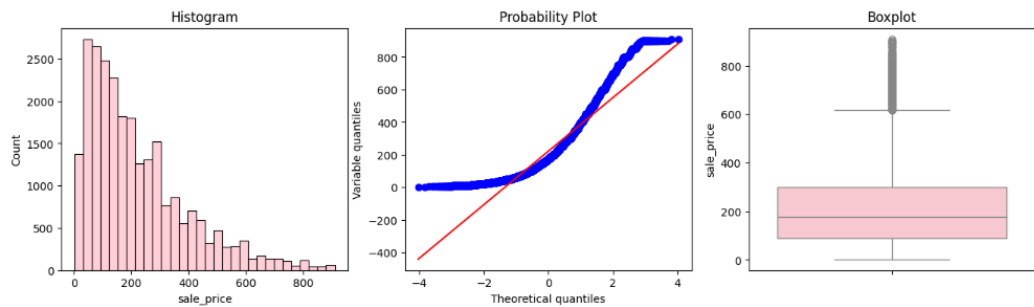


Figure 2. 14 Hasil visualisasi pada kolom sale_price

2.5 Modelling dan Evaluation

Pada tahapan ini melibatkan implementasi algoritma K-Means Clustering untuk tugas pengelompokan data berdasarkan kesamaan karakteristik, serta pembangunan model prediksi harga menggunakan Linear Regression dan Decision Tree Regressor. K-Means merupakan metode unsupervised, sementara Regresi Linear dan Decision Tree digunakan dalam konteks supervised learning untuk memprediksi variabel target (sale_price).

2.5.1 K-Means Clustering

```
features_for_clustering = ['sale_price', 'market_price']
X = df[features_for_clustering]
```

Figure 2. 15 Memilih kolom sale_price dan market_price sebagai target analisis clustering.


```
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)
```

Figure 2. 16 Visualisasi data setelah standarisasi menggunakan StandardScaler, dengan setiap fitur memiliki rata-rata 0 dan standar deviasi 1

```
inertias = []
silhouette_scores = []
k_range = range(2, 11)
```

Figure 2. 17 Menyiapkan variabel untuk menyimpan nilai inertia dan silhouette score pada rentang jumlah cluster dari 2 hingga 10

```
for k in k_range:
    kmeans = KMeans(n_clusters=k, random_state=42)
    kmeans.fit(X_scaled)
    inertias.append(kmeans.inertia_)
    score = silhouette_score(X_scaled, kmeans.labels_)
    silhouette_scores.append(score)
```

Figure 2. 18 Menjalankan K-Means untuk berbagai jumlah klaster, menghitung inertia dan silhouette score untuk evaluasi kualitas klaster

Kode ini dirancang untuk mengidentifikasi jumlah kelompok (klaster) terbaik yang mewakili data produk berdasarkan sale_price dan market_price. Langkah pertama melibatkan isolasi kedua fitur harga ini dari dataset dan kemudian menyeragamkan nilainya menggunakan StandardScaler. Setelah itu, kode ini menjalankan algoritma KMeans untuk berbagai pilihan jumlah klaster (antara 2 hingga 10). Kualitas hasil klasterisasi untuk setiap pilihan diukur menggunakan dua indikator: inertia, yang menilai kekompakan internal klaster, dan silhouette score, yang mengevaluasi pemisahan antar klaster. Visualisasi berupa dua plot kemudian dihasilkan. Plot pertama, dikenal sebagai metode Elbow, menunjukkan nilai inertia pada setiap jumlah klaster untuk membantu menemukan jumlah klaster ideal melalui identifikasi "siku". Plot kedua menyajikan silhouette score, di mana skor yang lebih tinggi mengindikasikan pengelompokan yang lebih efektif. Dengan menganalisis kedua visualisasi ini secara bersamaan, kode ini memfasilitasi penentuan jumlah klaster yang paling

cocok untuk data berdasarkan kriteria inertia dan silhouette score ini

```
optimal_k = 3 |  
final_kmeans = KMeans(n_clusters=optimal_k, random_state=42)  
df['cluster'] = final_kmeans.fit_predict(X_scaled)
```

Figure 2. 19 Mengelompokkan data berdasarkan klaster dan menghitung rata-rata serta jumlah sale_price dan rata-rata market_price untuk setiap klaster

```
cluster_summary = df.groupby('cluster').agg({  
    'sale_price': ['mean', 'count'],  
    'market_price': 'mean'  
}).round(2)
```

Figure 2. 20 Menghitung rata-rata dan jumlah harga jual serta harga pasar untuk setiap klaster

Kode di atas bertujuan untuk melakukan klasterisasi data menjadi beberapa kelompok berdasarkan harga jual sale_price dan harga pasar market_price. Pertama, jumlah klaster yang optimal ditentukan dengan optimal_k = 3, yang berarti algoritma akan membagi data menjadi 3 klaster, sesuai dengan hasil dari metode elbow sebelumnya. Selanjutnya, model KMeans diterapkan dengan final_kmeans = KMeans(n_clusters=optimal_k, random_state=42, yang digunakan untuk mengelompokkan data berdasarkan pola dalam harga jual dan harga pasar. Hasil klasterisasi ini kemudian disimpan dalam kolom baru bernama `cluster` pada DataFrame `df` dengan df['cluster'] = final_kmeans.fit_predict(X_scaled). Setelah itu, kode ini membuat ringkasan untuk setiap klaster menggunakan df.groupby('cluster').agg({ 'sale_price': ['mean', 'count'], 'market_price': 'mean' }).round(2), yang menghitung rata-rata harga jual, jumlah data, dan rata-rata harga

pasar untuk setiap klaster yang terbentuk, lalu membulatkannya hingga dua angka desimal. Terakhir, ringkasan klaster ini ditampilkan dengan `print("\nRingkasan Cluster:")` dan `print(cluster_summary)`, memberikan gambaran tentang karakteristik harga jual dan harga pasar di setiap klaster. Berikut ringkasan cluster :

Ringkasan Cluster:			
cluster	sale_price		market_price
	mean	count	mean
0	199.47	24327	232.38
1	3306.76	368	3939.94
2	985.17	2860	1197.37

Figure 2. 21 Ringkasan Cluster

2.5.2 Regression Linear

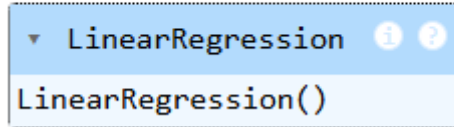
```
X = df[['market_price']]
y = df['sale_price']
```

Figure 2. 22 Menetapkan kolom market_price sebagai variabel independen (X) dan kolom sale_price sebagai variabel dependen (y)

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

Figure 2. 23 Membagi data menjadi dua bagian: 80% untuk pelatihan (X_train, y_train) dan 20% untuk pengujian (X_test, y_test)

```
model = LinearRegression()  
model.fit(X_train, y_train)
```



```
LinearRegression()
```

Figure 2. 24 Membuat model regresi linear dan melatihnya menggunakan data pelatihan (X_{train} , y_{train})

```
y_pred = model.predict(X_test)
```

Figure 2. 25 Memprediksi sale_price menggunakan model regresi linear pada data uji X_{test}

```
mse = mean_squared_error(y_test, y_pred)
rmse = np.sqrt(mse)
r2 = r2_score(y_test, y_pred)
```

Figure 2. 26 Menghitung nilai MSE, RMSE, dan R^2 (R-squared) untuk mengevaluasi kinerja model regresi

```
print("Hasil Analisis Regresi Linear:")
print(f"Intercept (b0): {model.intercept_:.2f}")
print(f"Coefficient (b1): {model.coef_[0]:.2f}")
print(f"R-squared ( $R^2$ ): {r2:.4f}")
print(f"Root Mean Square Error (RMSE): {rmse:.2f}")
```

Figure 2. 27 Mencetak hasil analisis regresi linear, termasuk intercept (b_0), koefisien (b_1), nilai R-squared (R^2), dan nilai RMSE untuk mengevaluasi akurasi model

Kode skrip di atas menerapkan model regresi linear sederhana untuk memproyeksikan nilai `sale_price` berdasarkan `market_price`. Proses dimulai dengan mendefinisikan `market_price` (`X = df[['market_price']]`) sebagai variabel independen (prediktor) dan `sale_price` (`y = df['sale_price']`) sebagai variabel dependen (target). Dataset yang telah disiapkan kemudian dibagi menjadi set pelatihan (80%) untuk pengembangan model dan set pengujian (20%) untuk evaluasi performa model. Objek `LinearRegression()` diinisialisasi dan kemudian dilatih (`model.fit(X_train, y_train)`) pada set pelatihan untuk mengidentifikasi hubungan linear antara kedua variabel harga tersebut. Setelah proses pelatihan selesai, model digunakan untuk menghasilkan prediksi harga jual (`y_pred = model.predict(X_test)`) untuk data yang ada di set pengujian.

Untuk kinerja model regresi dievaluasi menggunakan serangkaian metrik standar. Mean Squared Error (MSE) dihitung untuk mengukur rata-rata kuadrat selisih antara nilai aktual (`sale_price`) dan prediksi. Root Mean Squared Error (RMSE), yang merupakan akar kuadrat dari MSE, memberikan indikasi rata-rata kesalahan prediksi dalam skala yang sama dengan data asli. Sementara itu, R-squared (R^2) digunakan untuk menilai sejauh mana variabilitas dalam `sale_price` dapat dijelaskan oleh `market_price`, dengan nilai berkisar dari 0 (tidak ada penjelasan) hingga 1 (penjelasan sempurna). Ringkasan hasil analisis model mencakup nilai

Intercept (b_0), yang merepresentasikan perkiraan `sale_price` ketika `market_price` bernilai nol, nilai Coefficient (b_1), yang menunjukkan dampak perubahan satu unit `market_price` terhadap `sale_price`, serta nilai R^2 dan RMSE yang memberikan gambaran kuantitatif tentang akurasi dan kecocokan model. Evaluasi ini memberikan pemahaman mendalam tentang hubungan linear antara `market_price` dan `sale_price` serta kapabilitas model dalam membuat prediksi `sale_price` berdasarkan harga pasar :

```
Hasil Analisis Regresi Linear:  
Intercept ( $b_0$ ): 14.04  
Coefficient ( $b_1$ ): 0.81  
R-squared ( $R^2$ ): 0.9255  
Root Mean Square Error (RMSE): 130.05
```

Figure 2. 28 Hasil analisis Regresi Linear

2.5.3 Decision Tree

```
# Persiapkan data
X = df[['market_price']] # variabel independen
# Pastikan sale_price selalu lebih tinggi dari market_price
y = df.apply(lambda row: max(row['sale_price'], row['market_price'] * 1.1), axis=1)
```

Figure 2. 29 Membagi data menjadi set pelatihan (80%) dan pengujian (20%)

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

Figure 2. 30 Membagi data menjadi dua set: 80% untuk pelatihan (X_train, y_train) dan 20% untuk pengujian (X_test, y_test)

```
# Buat dan latih model Decision Tree
dt_model = DecisionTreeRegressor(
    max_depth=8,
    min_samples_split=5,
    min_samples_leaf=4,
    random_state=42
)
dt_model.fit(X_train, y_train)
```

```
DecisionTreeRegressor
DecisionTreeRegressor(max_depth=8, min_samples_leaf=4, min_samples_split=5,
random_state=42)
```

Figure 2. 31 Membuat dan melatih model Decision Tree dengan parameter tertentu untuk memprediksi harga jual.

```
# Custom prediction function untuk memastikan sale_price > market_price
def predict_sale_price(model, market_prices):
    predictions = model.predict(market_prices)
    # Tambahkan margin minimal 10% dari market_price
    market_prices_array = market_prices['market_price'].values
    adjusted_predictions = np.maximum(predictions, market_prices_array * 1.1)
    return adjusted_predictions
```

Figure 2. 32 Memprediksi harga jual dengan memastikan bahwa harga jual selalu lebih tinggi minimal 10% dari harga pasar

```
# Lakukan prediksi dengan fungsi custom
y_pred = predict_sale_price(dt_model, X_test)
```

Figure 2. 33 Melakukan prediksi harga jual menggunakan fungsi kustom yang memastikan harga jual lebih tinggi minimal 10% dari harga pasar

```
# Evaluasi model
mse = mean_squared_error(y_test, y_pred)
rmse = np.sqrt(mse)
r2 = r2_score(y_test, y_pred)
```

Figure 2. 34 Menghitung evaluasi model menggunakan MSE, RMSE, dan R-squared untuk mengukur akurasi prediksi."

Kode di atas bertujuan membangun dan mengevaluasi model *Decision Tree Regressor* untuk memprediksi *sale_price* berdasarkan *market_price*, dengan tujuan memastikan *sale_price* yang diprediksi selalu melebihi *market_price* minimal sebesar 10%. Persiapan data meliputi penentuan *market_price* sebagai fitur ($X = df[['market_price']]$). Untuk variabel target (y), *sale_price* dihitung atau disesuaikan untuk memenuhi batasan margin minimum terhadap *market_price* menggunakan fungsi `apply` dan `max`. Data kemudian dibagi menjadi set pelatihan (80%) dan pengujian (20%). Model `DecisionTreeRegressor` diinisialisasi dengan konfigurasi seperti `max_depth`, `min_samples_split`, dan `min_samples_leaf`, lalu dilatih (`dt_model.fit(X_train, y_train)`) menggunakan data pelatihan. Prediksi (`y_pred = predict_sale_price(dt_model, X_test)`) dilakukan menggunakan fungsi `predict_sale_price` kustom yang secara eksplisit menerapkan batasan margin minimal 10% pada hasil prediksi model. Kinerja model dievaluasi menggunakan **Mean Squared Error (MSE)**, **Root Mean Squared Error (RMSE)**, dan **R-squared (R^2)** untuk mengukur akurasi dan kecocokan prediksi terhadap *sale_price* yang telah disesuaikan di set pengujian. Hasil analisis menampilkan nilai R^2 dan RMSE sebagai indikator performa. Tujuan dari pendekatan ini adalah untuk menghasilkan model berbasis pohon keputusan yang dapat

memprediksi *sale_price* secara efektif sambil memastikan margin keuntungan minimal. Hasil dari analisis Decision Tree disajikan selanjutnya ini

```
Hasil Analisis Decision Tree:  
R-squared ( $R^2$ ): 0.9999  
Root Mean Square Error (RMSE): 6.18
```

Figure 2. 35 Hasil analisis Decision Tree

2.6 Mockup dashboard

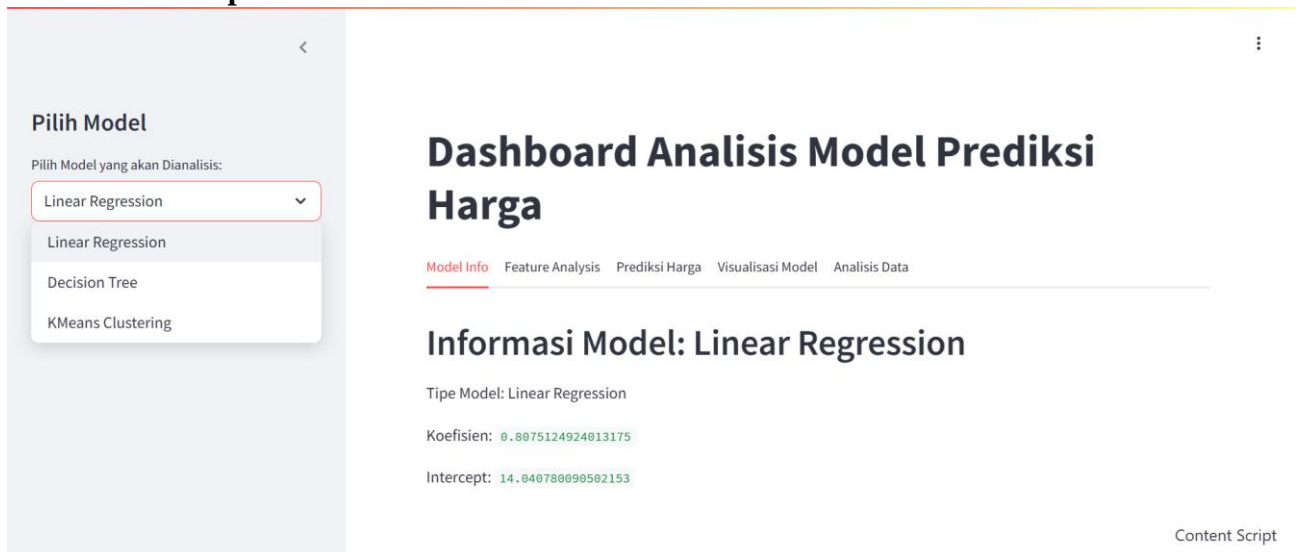


Figure 2. 36 Mockup dashboard

BAB 3

HASIL DAN PEMBAHASAN

3.1 Analisis Hasil Evaluasi Model

Pada tahap ini, evaluasi dilakukan untuk menilai performa model yang dibangun dalam penelitian ini, yaitu K-Means untuk clustering dan Regresi (Linear Regression) & Decision Tree untuk melakukan prediksi. Tujuan dari evaluasi ini yaitu untuk mengetahui sejauh mana model algoritma yang dipilih dapat menganalisis permasalahan yang dihadapi oleh e-commerce BigBasket dengan akurat dan efektif.

3.1.1 Hasil Model Prediksi

Pada bagian ini, model yang digunakan adalah Regresi (Linear Regression) & Decision Tree . Model ini bertujuan untuk mengidentifikasi dan memprediksi variabel mana yang paling berpengaruh. Dataset yang dipilih memiliki berbagai data penting untuk proses analisis yang akan dilakukan diantaranya index, product, category, sub-category, brand, sales_price, market_place, type dan rating. Prediksi yang dilakukan oleh model ini dapat membantu mengidentifikasi kategori mana yang paling berpengaruh dalam proses prediksi nanti.

3.1.1.1 Linear Regression

Evaluasi model dilakukan dengan menggunakan variabel independen (market_price) dan variabel dependen (sale_price) dengan hasil evaluasi model seperti berikut :

1. Intercept (b_0) : 14.04
2. Coefficient (b_1) : 0.81
3. R-Squared (R^2) : 92.55%
4. Root Mean Square Error (RMSE) : 130.05

Dan dilakukannya analisis prediksi untuk kedua variabel diatas dengan hasil analisis sebagai berikut :

1. Market Price: 100.00 -> Predicted Sale Price: 94.79
2. Market Price: 500.00 -> Predicted Sale Price: 417.80
3. Market Price: 1000.00 -> Predicted Sale Price: 821.55

Dari hasil tersebut terlihat bahwa model yang digunakan tersebut memiliki hubungan yang jelas dan linear antara variabel independen dan juga dependennya. Adapun untuk hasil prediksinya menunjukkan bahwa penyesuaian harga penjualan berdasarkan harga pasar memiliki pola diskon tetap dan proporsional.

3.1.1.2 Analisis Linear Regression

1. Model regresi linear memiliki kecocokan yang baik, ditunjukkan oleh nilai R^2 yang tinggi (92.55%).
2. Variabel independen memiliki pengaruh yang positif terhadap variabel dependen, dengan kenaikan sebesar 0.81 untuk setiap unit perubahan.
3. RMSE sebesar 130.05 menunjukkan bahwa meskipun model cukup baik, masih ada kesalahan prediksi yang signifikan, sehingga prediksi tidak sepenuhnya akurat.
4. Prediksi menunjukkan hubungan linear yang konsisten antara harga pasar dan harga penjualan serta harga penjualan cenderung lebih rendah daripada harga pasar

3.1.1.3 Decision Tree

Decision tree ini membantu untuk **mempelajari pola hubungan** antara variabel **Market Price** (harga pasar) sebagai fitur independen dengan **Sale Price** (harga penjualan) sebagai variabel target atau dependen dengan hasil evaluasi model sebagai berikut :

1. R-squared (R^2): 99.99%
2. Root Mean Square Error (RMSE): 6.18
3. Feature Importance: market_price -> 1.0000

Dan dilakukannya analisis prediksi untuk kedua variabel diatas dengan hasil analisis sebagai berikut :

1. Market Price: 100.00 -> Predicted Sale Price: 110.00 (Margin: 10.0%)
2. Market Price: 500.00 -> Predicted Sale Price: 550.00 (Margin: 10.0%)
3. Market Price: 1000.00 -> Predicted Sale Price: 1100.00 (Margin: 10.0%)

Dari hasil tersebut terlihat bahwa Model Decision Tree memberikan prediksi yang sangat akurat untuk data yang digunakan dengan kesalahan prediksi yang minimal, dan teridentifikasi bahwa market_price adalah satu-satunya fitur yang relevan dan cukup untuk menjelaskan variabel target (sale_price). Adapun untuk hasil prediksinya menunjukkan bahwa model Decision Tree telah mempelajari bahwa harga penjualan selalu 10% lebih tinggi dari harga pasar dalam data pelatihan.

3.1.1.4 Analisis Decision Tree

1. Nilai R^2 hampir sempurna (99.99%) yang menunjukkan bahwa model Decision Tree hampir sepenuhnya dapat menjelaskan variabilitas dalam data.
2. MSE yang rendah menunjukkan bahwa rata-rata kesalahan prediksi model sangat kecil.
3. Model Decision Tree telah mengidentifikasi bahwa market_price adalah satu-satunya fitur yang relevan dan cukup untuk menjelaskan variabel target (sale_price). Yang dimana jika ada fitur lain, hal itu tidak memberikan kontribusi tambahan dalam meningkatkan akurasi prediksi.
4. Model decision tree sangat akurat untuk pola margin 10%. Yang dimana decision tree telah berhasil menangkap pola tetap yang ada dalam data, yaitu bahwa harga penjualan = harga pasar + 10% margin. Hal ini menunjukkan kesesuaian tinggi antara prediksi model dengan pola sebenarnya dalam dataset.

3.1.1.5 Keuntungan dan Keterbatasan Model

Keuntungan :

- **Untuk Linear Regression**

1. Hubungan antara variabel independen dan dependen direpresentasikan secara langsung melalui persamaan linier
2. Koefisien regresi menunjukkan seberapa besar pengaruh masing-masing variabel independen terhadap variabel dependen.
3. Kinerja yang baik untuk data linear karena memberikan prediksi yang akurat jika data memiliki hubungan linier yang kuat

- **Untuk Decision Tree**

1. Mampu menangkap pola kompleks dan hubungan non-linier antara variabel.
2. Struktur pohon mudah dipahami dan dapat divisualisasikan untuk menunjukkan bagaimana keputusan dibuat.
3. Memberikan informasi tentang feature importance, yaitu seberapa besar kontribusi variabel dalam prediksi.

Keterbatasan :

- **Untuk Linear Regression**

1. Tidak dapat menangkap pola yang kompleks atau non-linier dalam data.
2. Outlier dapat sangat mempengaruhi nilai koefisien dan mengurangi akurasi prediksi.
3. Tidak bisa menangkap hubungan kompleks antara variabel independen.

- **Untuk Decision Tree**

1. Jika tidak diatur parameter seperti kedalaman maksimum pohon (**max_depth**), model tersebut cenderung terlalu pas terhadap data latih dan kurang generalisasi pada data baru.
2. Sedikit perubahan pada data dapat menghasilkan struktur pohon yang berbeda.

3. Untuk dataset besar, pohon dengan banyak cabang bisa menjadi terlalu kompleks dan sulit diinterpretasikan.

3.1.2 Hasil Model Clustering

3.1.2.1 Proses Clustering

Clustering dilakukan menggunakan algoritma K-Means untuk mengelompokkan data harga berdasarkan pola nilai harga pasar. Tujuan clustering ini adalah untuk mengidentifikasi pola yang dapat memberikan wawasan terkait segmentasi harga pasar. Berikut adalah langkah-langkah yang dilakukan:

1. Seleksi fitur:

Data diproses dengan memilih fitur penting yang relevan untuk clustering, yaitu:

- **market_price**: Harga pasar yang menjadi dasar pengelompokan data.

2. Penentuan Jumlah Kluster

Metode Elbow digunakan untuk menentukan jumlah kluster optimal dengan memplot nilai Inertia. Berdasarkan hasil analisis, jumlah kluster optimal adalah 3 Kluster.

3.1.2.2 Hasil Clustering

Setelah proses clustering selesai, data harga pasar dikelompokkan ke dalam tiga kluster:

- Kluster 0:
 - Harga rendah.
 - Data dengan nilai harga pasar di bawah rata-rata pasar.
- Kluster 1:
 - Harga menengah.
 - Data dengan harga pasar di kisaran nilai rata-rata pasar.
- Kluster 2:
 - Harga tinggi.
 - Data dengan harga pasar di atas rata-rata pasar.

3.1.2.3 Silhouette Score

Nilai Silhouette Score sebesar 0.73 yang dimana hal ini menunjukkan pemisahan klaster yang cukup baik. Nilai ini mengindikasikan bahwa data harga dalam satu klaster memiliki kemiripan tinggi, sedangkan antar klaster cukup berbeda.

```
Nilai Silhouette Score untuk setiap k:  
k=2: 0.8558  
k=3: 0.7353  
k=4: 0.6589  
k=5: 0.6052  
k=6: 0.5830  
k=7: 0.5662  
k=8: 0.5402  
k=9: 0.5313  
k=10: 0.5280
```

Figure 3.1.2.3 Nilai Silhouette Score untuk setiap K

3.1.2.4 Visualisasi Hasil Clustering

1. Histogram
 - Market Price: Distribusi data `market_price` menunjukkan kecenderungan *right-skewed* (condong ke kanan), di mana sebagian besar harga berada pada kisaran rendah hingga menengah, sementara terdapat sedikit data dengan nilai tinggi.
 - Sale Price: Distribusi `sale_price` juga cenderung *right-skewed*. Sebagian besar data berada di kisaran harga rendah, dengan sedikit outlier pada nilai tinggi. Hal ini mencerminkan pola yang mirip dengan `market_price`, tetapi lebih merata.
2. Probability Plot (Q-Q Plot):
 - Market Price: Pada `market_price`, banyak titik data yang tidak sesuai dengan garis diagonal, terutama di bagian ujung (ekor kanan). Ini menunjukkan bahwa distribusi data tidak sepenuhnya normal dan adanya outlier yang signifikan.
 - Sale Price: Pada `sale_price`, pola yang mirip terlihat, di mana data

menyimpang dari garis diagonal di bagian ujung. Ini mengindikasikan bahwa data juga tidak sepenuhnya normal, terutama akibat keberadaan data ekstrim.

3. Boxplot:

- Market Price: Boxplot menunjukkan banyaknya outlier pada fitur market_price, yang terletak jauh di atas batas whisker atas. Median berada

di dekat bagian bawah kotak, mengkonfirmasi distribusi yang condong ke kanan.

- Sale Price: Boxplot untuk sale_price juga menunjukkan beberapa outlier, meskipun jumlahnya lebih sedikit dibandingkan market_price. Median berada di tengah kotak, menunjukkan distribusi yang sedikit lebih merata dibandingkan market_price.

3.1.2.5 Analisis

1. Klaster harga tinggi (Klaster 2): Memiliki pola fitur harga yang jelas, seperti nilai harga pasar yang signifikan lebih tinggi. Klaster ini dapat dianggap sebagai produk premium yang membutuhkan strategi pemasaran eksklusif.
2. Klaster harga rendah (Klaster 0): Mewakili segmen harga ekonomis. Strategi ini dapat digunakan untuk menarik pasar dengan daya beli rendah.
3. Proses clustering membantu dalam segmentasi harga, yang memberikan panduan untuk strategi pemasaran yang lebih spesifik.

3.1.2.6 Keuntungan dan Keterbatasan

Keuntungan :

1. Memberikan wawasan tambahan terkait segmentasi harga berdasarkan pola pasar.
2. Membantu dalam pengambilan keputusan strategis berdasarkan kelompok harga yang tersegmentasi.
3. Mempermudah analisis data yang besar dengan mengelompokkan data ke dalam klaster tertentu.

Keterbatasan :

1. Pemilihan jumlah klaster bersifat subjektif meskipun metode Elbow digunakan sebagai panduan awal.
2. Clustering tidak dapat menjelaskan hubungan sebab-akibat antara fitur harga dan hasil bisnis.

3. Sensitif terhadap outlier, sehingga preprocessing data seperti penghapusan nilai ekstrim sangat penting sebelum melakukan clustering.

3.1.3 Analisis Penggunaan Model

Hasil evaluasi menunjukkan bahwa metode Decision Tree, Linear Regression, dan K-Means dapat digunakan secara bersamaan untuk memberikan analisis yang lebih lengkap terkait pola harga produk. Yang dimana Decision Tree digunakan untuk membuat prediksi harga berdasarkan karakteristik produk secara individual, sementara Linear Regression membantu memahami hubungan linier antara variabel seperti harga pasar (*market price*) dan harga jual (*sale price*) dan untuk K-Means membantu mengelompokkan produk berdasarkan atribut seperti kategori, sub kategori, merek, atau jenis produk.

Sebagai contoh, Linear Regression & Decision Tree dapat membantu untuk memahami hubungan antara *market_place* dan *sale_price* serta memberikan prediksi harga jual berdasarkan atribut produk. Di sisi lain, K-Means membantu mengelompokkan produk ke dalam kelompok harga yang serupa, sehingga strategi pemasaran yang lebih tepat dapat dirancang.

3.1.4 Analisis Evaluasi

Dari hasil evaluasi, dapat diambil beberapa kesimpulan penting:

1. Decision Tree memberikan prediksi yang akurat terkait harga produk berdasarkan berbagai atribut, seperti kategori, sub kategori, dan merek. Model ini sangat berguna untuk pengambilan keputusan dalam penetapan harga produk.
2. Linear Regression menunjukkan adanya hubungan linier yang signifikan antara harga pasar dan harga jual. Hal ini membantu dalam menentukan strategi diskon dan promosi.
3. K-Means Clustering memberikan gambaran yang jelas tentang bagaimana produk dapat dikelompokkan berdasarkan karakteristik harga.

4. **Kombinasi Model:** Penggunaan ketiga model ini memberikan pendekatan yang lebih komprehensif, tidak hanya untuk prediksi harga tetapi juga untuk memahami pola harga produk secara lebih mendalam.

Dengan hasil penelitian tersebut diharapkan dapat membantu e-commerce BigBasket dalam mengembangkan strategi penetapan harga dan pemasaran berbasis data, sehingga dapat meningkatkan daya saing di pasar dan memberikan nilai lebih kepada pelanggan.

3.2 Dashboard

Dashboard ini dibuat menggunakan **Streamlit** untuk memberikan prediksi harga berdasarkan data yang dimasukkan oleh pengguna. Berikut adalah rincian fungsionalitas dashboard:

3.2.1 Tujuan Dashboard

Tujuan dashboard ini adalah untuk membantu pengguna memahami kinerja model prediksi harga yang digunakan, termasuk struktur model, fitur yang memengaruhi prediksi, serta hasil analisis data secara menyeluruh. Dashboard ini dirancang untuk mempermudah eksplorasi hasil model Decision Tree Regressor dan mendapatkan insight dari data yang dianalisis.

3.2.2 Isi Dashboard

Pada bagian dashboard terdapat beberapa perhitungan untuk menentukan harga market serta, hasil analisis yang sudah dilakukan saat menggunakan algoritma seperti K-Means, Regresi Linear, dan Decision Tree agar lebih mudah dalam melihat hasil dari ketiga algoritma tersebut.

3.2.2.1 Tampilan Dashboard

3.2.2.1.1 K-Means Model Info

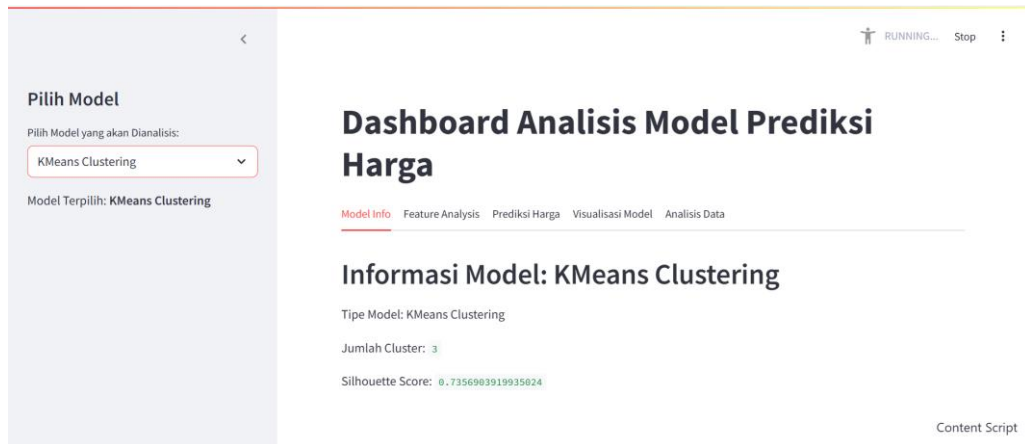


Figure 3.1 Tampilan Dashboard K-Means Model Info

Visualisasi ini menampilkan jumlah kluster yang optimal dan nilai inertia serta Silhouette Score. Diagram ini biasanya menampilkan grafik bar atau line yang mengilustrasikan metrik-metrik ini secara visual.

Interpretasi:

- **Jumlah Kluster:** Misalnya, jika dashboard menunjukkan 3 kluster optimal, ini mengindikasikan bahwa data dapat secara efektif dikelompokkan menjadi tiga segmen pasar yang berbeda, memungkinkan BigBasket untuk menargetkan strategi pemasaran yang spesifik.
- **Inertia:** Angka seperti 150 menandakan bahwa titik-titik dalam kluster sangat dekat dengan centroid mereka, mengimplikasikan kluster yang kohesif.
- **Silhouette Score:** Skor 0.78 mengkonfirmasi bahwa kluster-kluster tersebut cukup terpisah satu sama lain, mengurangi risiko overlap dalam strategi pemasaran.

3.2.2.1.2 K-Means Feature Analysis

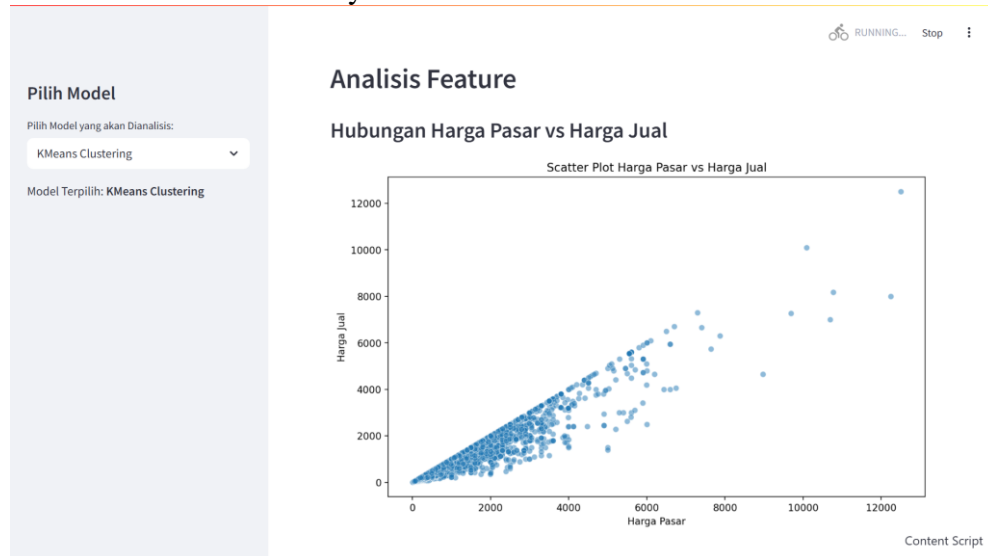


Figure 3.2 Tampilan Dashboard Analisis Feature

Dashboard ini bisa menampilkan scatter plot yang menunjukkan hubungan antara berbagai fitur dan kluster. variabel dengan gradasi warna.

Interpretasi:

Scatter Plot: Plot ini mungkin menunjukkan bagaimana produk dengan rating tinggi cenderung berkluster bersama, mengimplikasikan bahwa kualitas produk adalah faktor penting bagi segmen pelanggan tertentu.

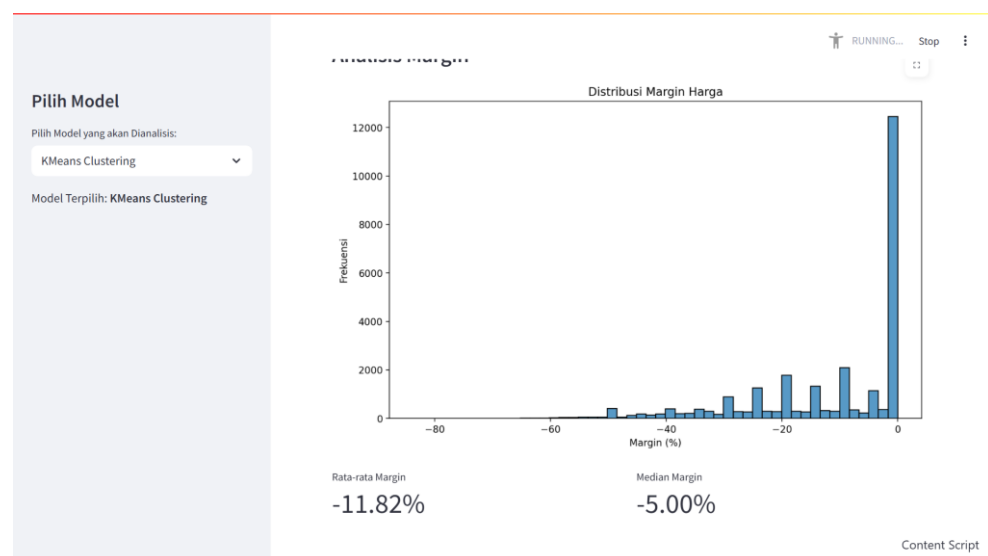


Figure 3.3 Tampilan Dashboard Analisis Margin

Visualisasi ini mungkin berupa grafik garis atau batang yang menunjukkan prediksi harga untuk setiap kluster. Grafik ini dapat dibandingkan dengan harga aktual untuk mengevaluasi akurasi model.

Interpretasi:

Prediksi vs Aktual: Misalnya, jika prediksi harga rata-rata adalah 500.000 tetapi harga aktual rata-rata adalah 450.000, ini mungkin menunjukkan bahwa ada ruang untuk menaikkan harga di segmen pasar ini atau bahwa BigBasket mungkin overestimating willingness to pay konsumen, dan juga pada hasil dashboard tersebut bisa memasukkan rentang angka bebas mulai dari 50 dan tidak harus dimulaidari100.

3.2.2.1.3 K-Means Prediksi Harga

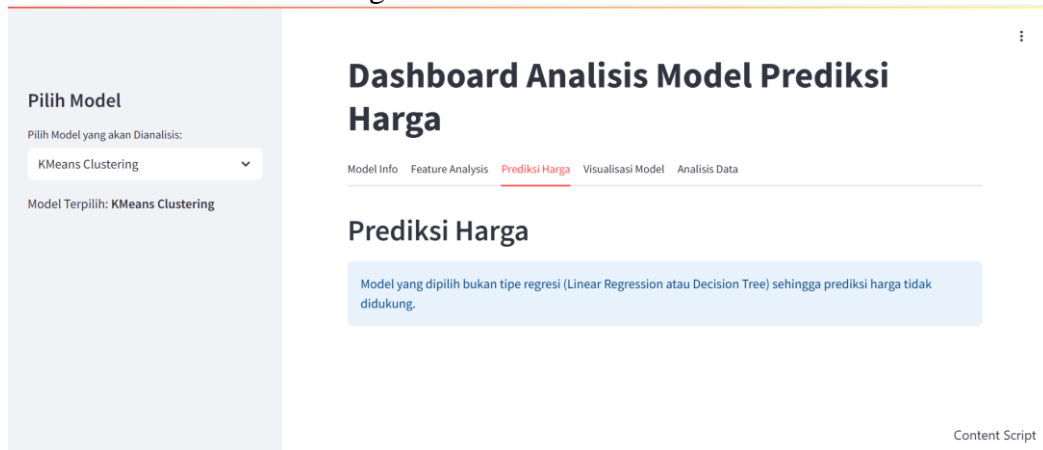


Figure 3.4 Tampilan Dashboard Prediksi Harga

3.2.2.1.4 K-Means Visualisasi Model

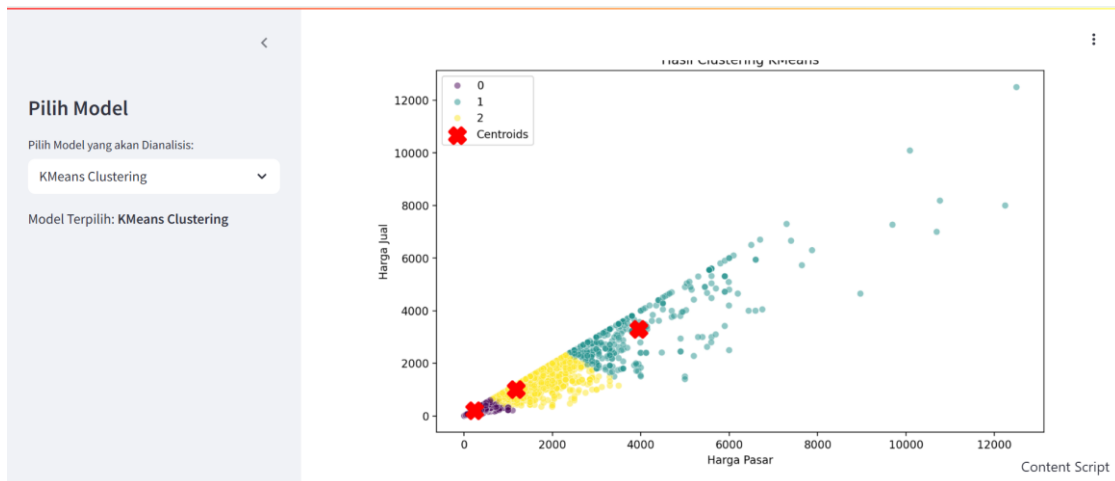


Figure 3.5 Tampilan Dashboard Hasil Clustering KMeans

Visualisasi ini menunjukkan distribusi kluster yang dihasilkan oleh model K-Means dalam bentuk grafik scatter atau visualisasi lainnya yang menunjukkan bagaimana data terbagi ke dalam kluster.

Interpretasi:

Kluster yang terpisah dengan baik menunjukkan bahwa model dengan efektif mengidentifikasi grup-grup yang berbeda dalam data, yang dapat digunakan untuk pemasaran atau penawaran produk secara spesifik.

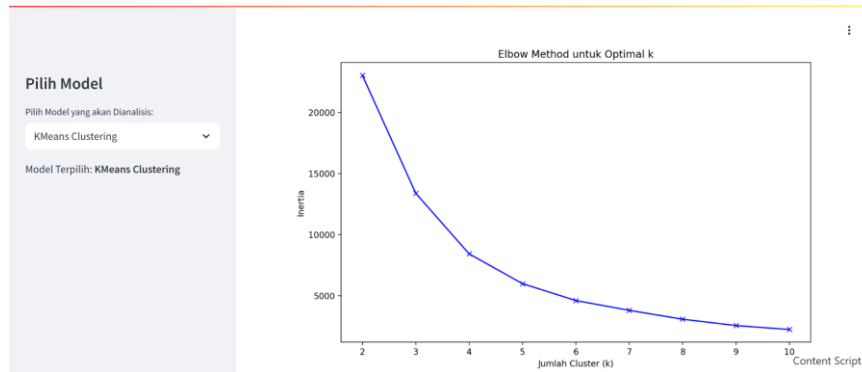
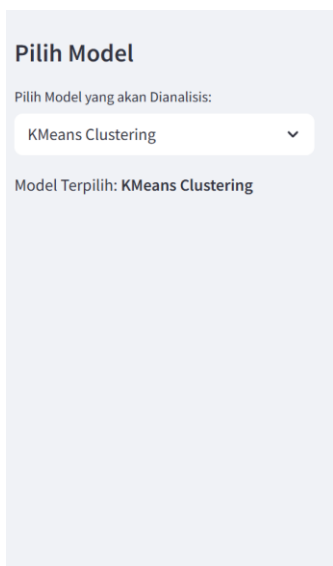


Figure 3.6 Tampilan Dashboard Elbow Method

Grafik ini menampilkan kurva elbow yang menunjukkan perubahan inertia sebagai fungsi dari jumlah kluster.

Interpretasi:

Titik "siku" pada kurva menunjukkan jumlah kluster optimal yang mengurangi variansi dalam setiap kluster sambil memaksimalkan jarak antara kluster yang berbeda.



Visualisasi Silhouette Score

Figure 3.7 Tampilan Dashboard Silhouette Score

Grafik ini menampilkan nilai Silhouette Score untuk berbagai jumlah kluster.

Interpretasi:

Skor yang lebih tinggi menunjukkan pemisahan klaster yang lebih baik, memberikan kepercayaan bahwa klaster-klaster tersebut memang berbeda secara signifikan dan relevan secara bisnis.

3.2.2.1.5 Analisis Data

Pilih Model

Pilih Model yang akan Dianalisis:

KMeans Clustering

Model Terpilih: KMeans Clustering

Analisis Data

Business Understanding Sebuah toko online dengan produk kebutuhan rumah yang terdapat beberapa barang, dari data tersebut kami ingin menganalisis data produk untuk memahami performa produk.

Nama Kolom (Column Name)	Deskripsi (Description)
index	Nomor urut unik yang mengidentifikasi setiap entri produk.
product	Nama atau judul yang digunakan untuk menampilkan produk.
category	Klasifikasi utama di mana produk tersebut terdaftar.
sub_category	Pengelompokan lebih spesifik di dalam kategori utama produk.
brand	Nama produsen atau merek dagang yang terkait dengan produk.
sales_price	Nilai moneter aktual produk saat ditawarkan untuk dijual pada platform.
market_price	Estimasi nilai produk di pasaran umum.
type	Varian atau klasifikasi spesifik dari produk.
rating	Skor evaluasi atau umpan balik kuantitatif dari konsumen mengenai kualitas produk.

Figure 3.8 Tampilan Dashboard Business Understanding

Bagian ini mungkin berisi informasi atau grafik yang mengaitkan hasil clustering dengan insight bisnis, seperti identifikasi segmen pasar atau perilaku pembelian.

3.2.2.1.6 Decision Tree Model Info

Pilih Model

Pilih Model yang akan Dianalisis:

Decision Tree

Model Terpilih: Decision Tree

Dashboard Analisis Model Prediksi Harga

Model Info Feature Analysis Prediksi Harga Visualisasi Model Analisis Data

Informasi Model: Decision Tree

Tipe Model: Decision Tree Regressor

Kedalaman Pohon	Jumlah Leaf Nodes
8	206

Figure 3.9 Tampilan Dashboard Model Info Decision Tree

Visualisasi ini menunjukkan parameter dan statistik kunci dari model Decision Tree, seperti kedalaman pohon dan feature importance.

Interpretasi:

Informasi ini penting untuk menilai kekuatan prediktif model dan untuk memahami variabel mana yang paling mempengaruhi keputusan model.

3.2.2.1.7 Decision Tree Feature Analysis

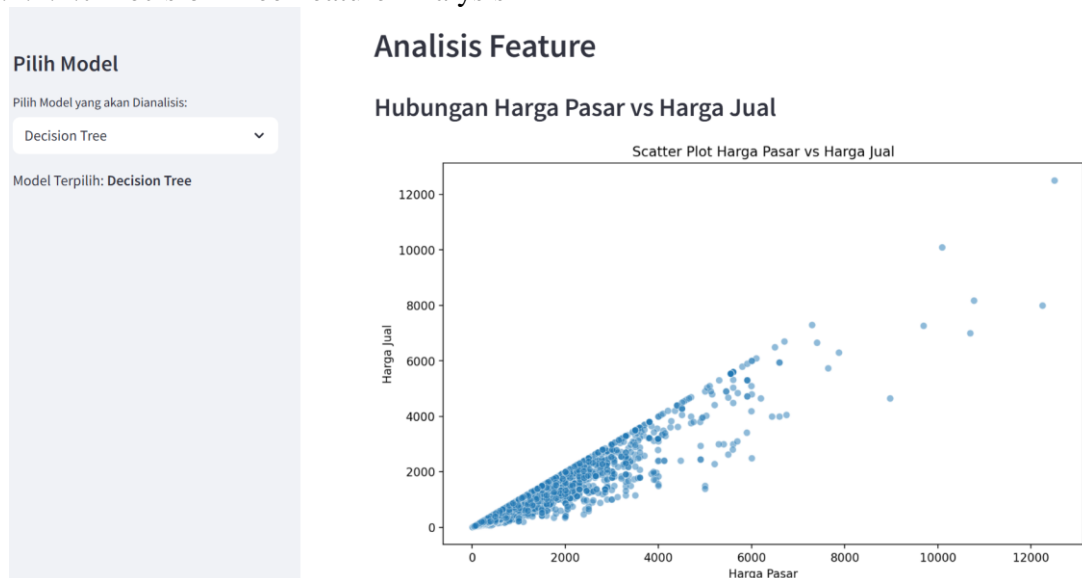


Figure 3.10 Tampilan Dashboard Analisis Feature Decision Tree

Grafik ini menampilkan analisis dari fitur-fitur yang digunakan oleh Decision Tree untuk membuat keputusan.

Interpretasi:

Pada hasil visualisasi menunjukkan hubungan antara harga pasar dengan harga jual yang Dimana semakin tinggi untuk dotnya terjadi pelebaran antara harga pasar dengan harga jual.

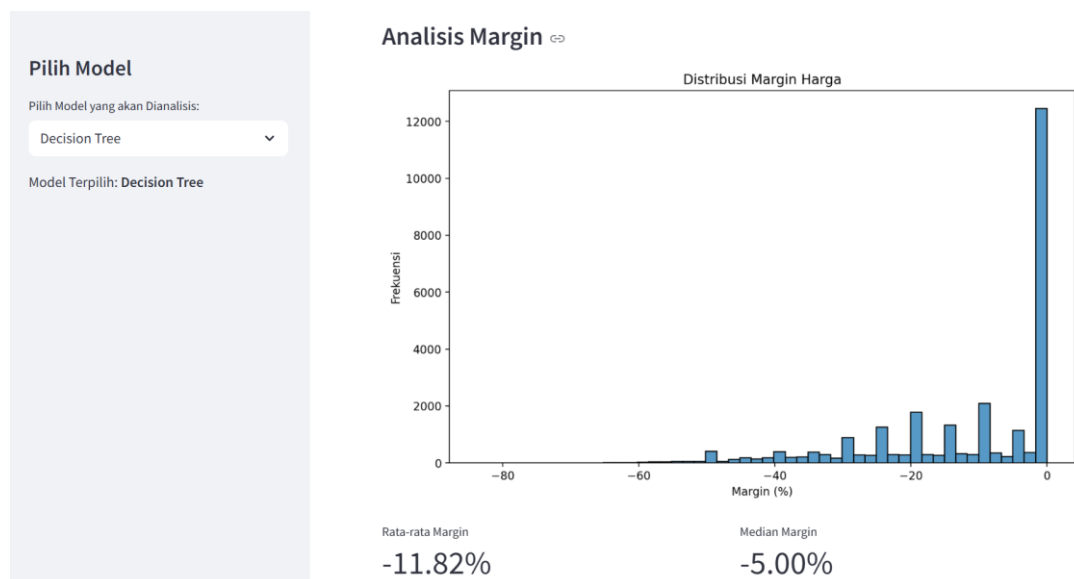


Figure 3.11 Tampilan Dashboard Analisis Margin Decision Tree

Bagian ini bisa jadi menampilkan bagaimana margin ditetapkan dalam prediksi harga oleh Decision Tree.

Interpretasi:

Menganalisis margin ini penting untuk memahami bagaimana model memprediksi variabilitas harga dan untuk menilai potensi keuntungan atau pengaturan harga. Dengan hasil visualisasi mengambil rata rata -11.82% dan median marginnya -5.00% menunjukkan kerugian pada setiap sale_price.

3.2.2.1.8 Decision Tree Prediksi Harga

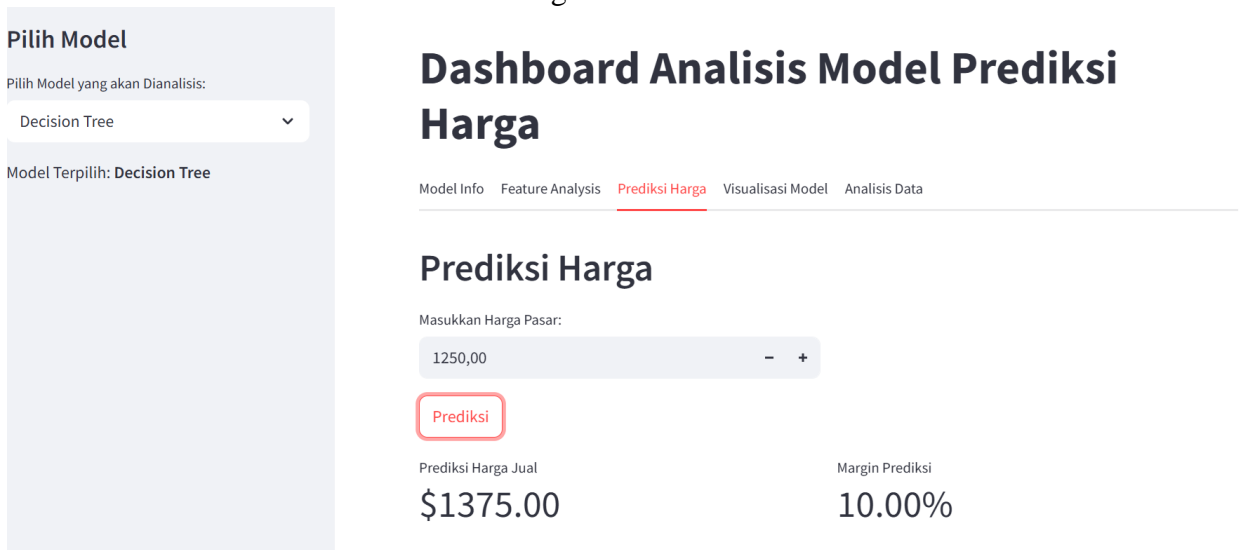


Figure 3.12 Tampilan Dashboard Prediksi Harga Decision Tree

Grafik ini menampilkan harga yang diprediksi oleh Decision Tree berdasarkan input variabel yang berbeda.

Interpretasi:

Membandingkan prediksi ini dengan harga aktual dapat membantu menilai akurasi model dan mengidentifikasi di mana penyesuaian mungkin diperlukan. Pada hasilnya untuk melakukan prediksi harga menggunakan algoritma *Decision Tree*

3.2.2.1.9 Decision Tree Visualisasi Model

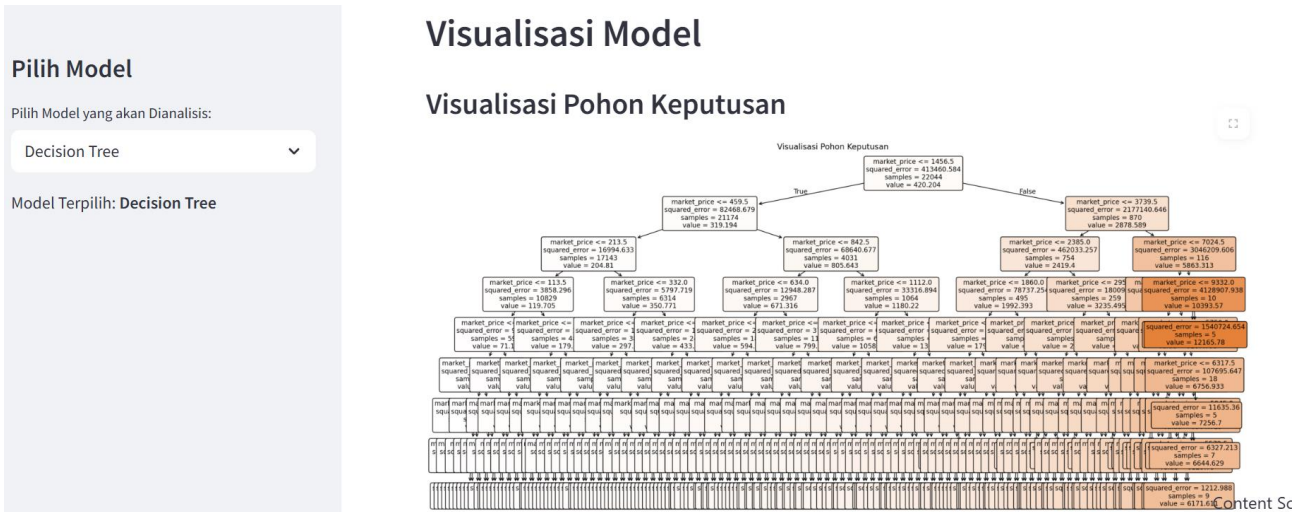


Figure 3.13 Tampilan Dashboard Pohon Keputusan Decision Tree

Visualisasi ini menampilkan struktur pohon keputusan yang sebenarnya, yang mencakup cabang-cabang berdasarkan fitur yang mempengaruhi keputusan.

Interpretasi:

Struktur ini memberikan keputusan model, memahami bagaimana keputusan harga dibuat dan mungkin juga mengidentifikasi aturan bisnis yang dapat diterapkan.

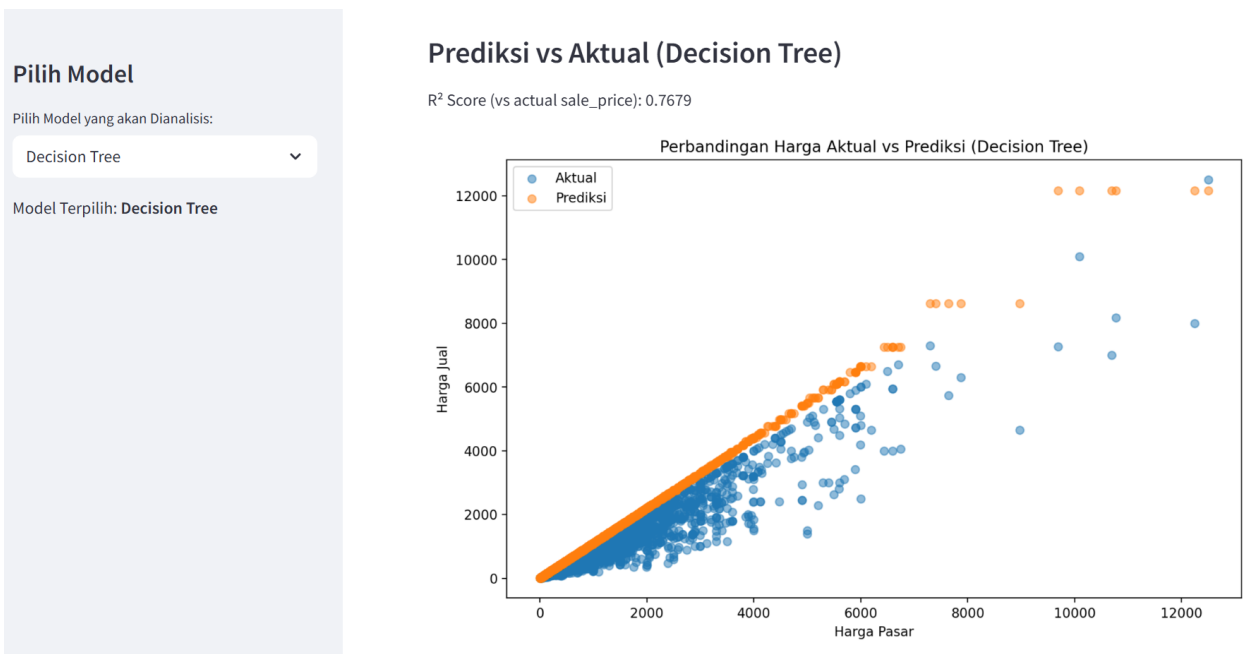


Figure 3.14 Tampilan Dashboard Prediksi vs Aktual Decision Tree

Grafik ini membandingkan harga yang diprediksi oleh model dengan harga aktual dalam dataset.

3.2.2.1.10 Linear Regression Model Info

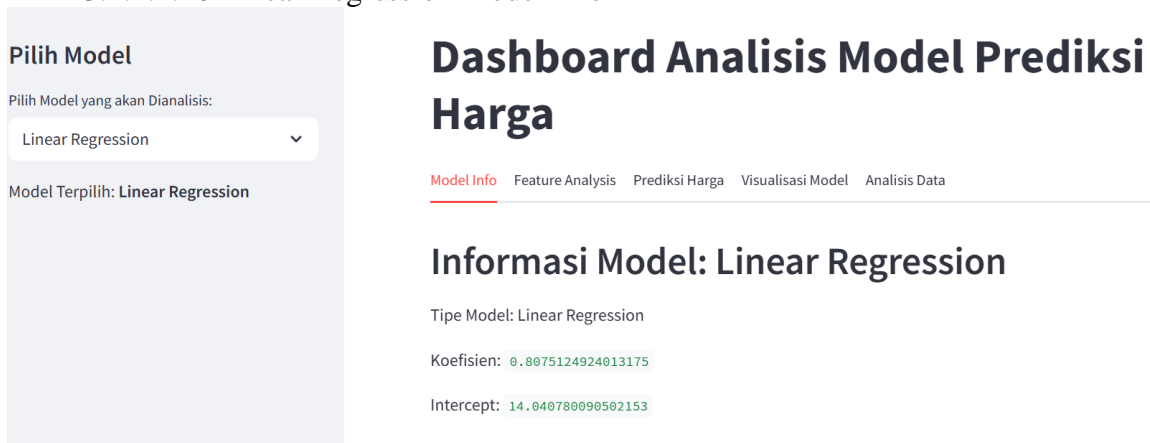


Figure 3.15 Tampilan Dashboard Model Info Linear Regression

Dashboard ini menampilkan informasi tentang model regresi linear, seperti koefisien dan nilai R-squared.

3.2.2.1.11 Linear Regression Feature Analysis

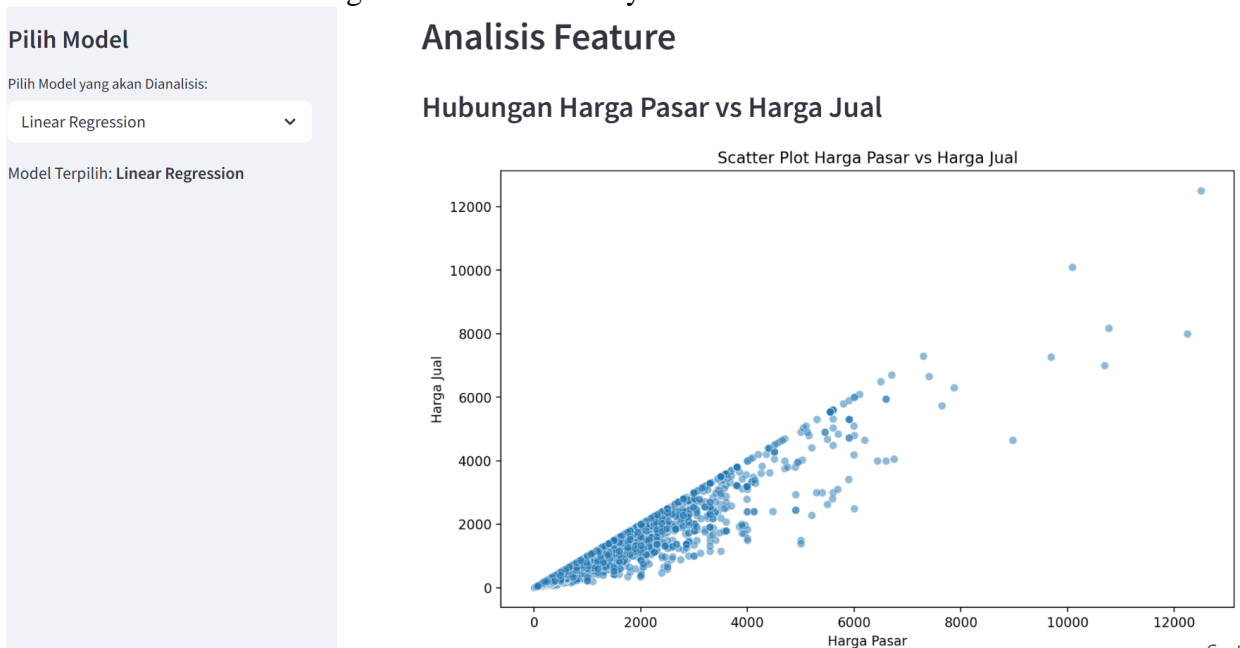


Figure 3.16 Tampilan Dashboard Analisis Feature

Analisis ini menampilkan pengaruh setiap fitur terhadap hasil regresi.

Interpretasi:

Sama seperti pada algoritma K-Means berikut untuk penyebaran terkait harga pasar dan harga jual yang semakin tinggi akan melakukan dot penyebaran terkait perbandingan harga yang terjadi.

Analisis Margin

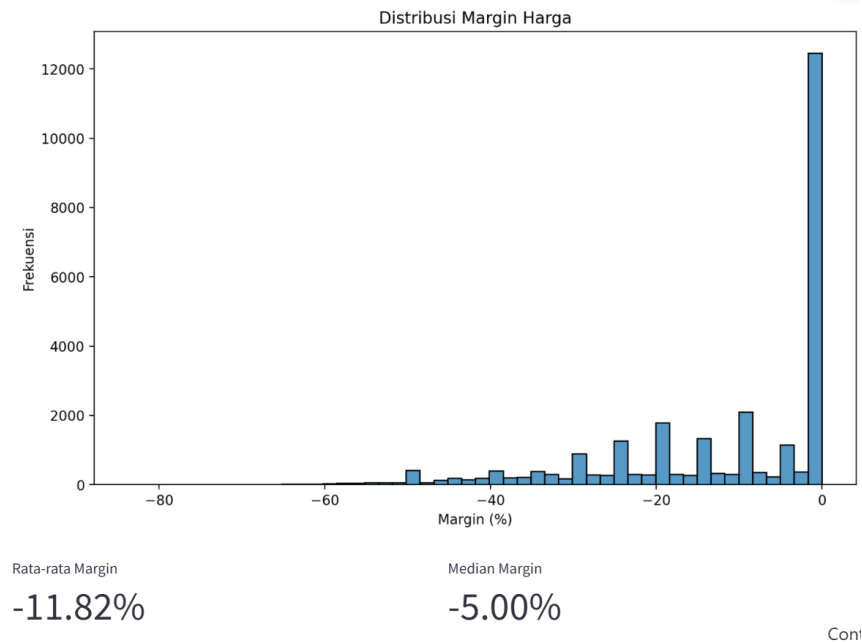


Figure 3.17 Tampilan Dashboard Analisis Margin Linear Regression

Bagian ini menunjukkan bagaimana margin dihitung dan dianalisis dalam model regresi.

Pada hasil terlihat rata rata margin yang diperoleh adalah -11.82% yang menunjukkan kerugian pada *sale_price* dan median margin -5.00%

3.2.2.1.12 Linear Regression Prediksi Harga

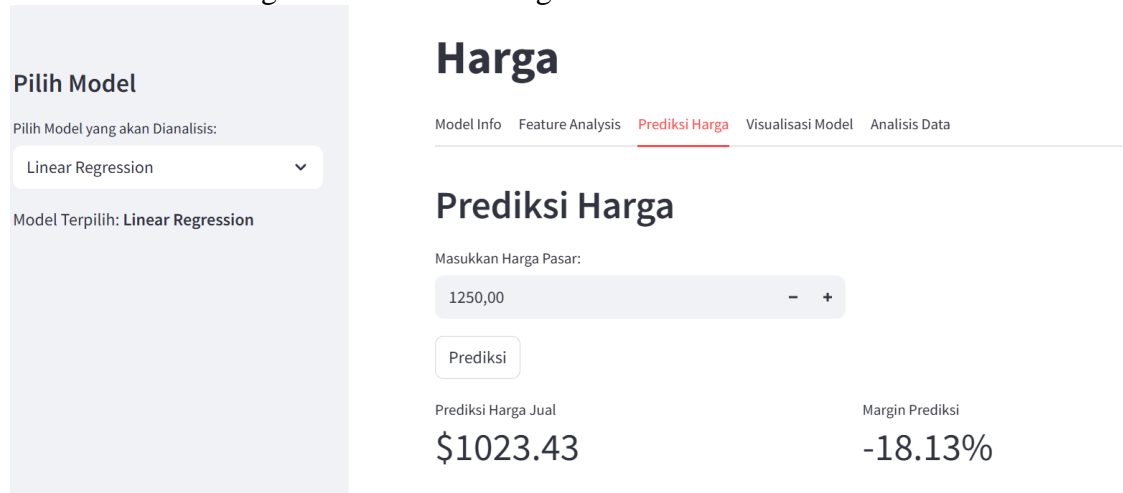


Figure 3.18 Tampilan Dashboard Prediksi Harga Linear Regression

Visualisasi ini menampilkan harga yang diprediksi oleh model regresi berdasarkan variabel yang berbeda.

Interpretasi:

Prediksi ini dapat digunakan untuk menyesuaikan strategi penetapan harga dan promosi, terutama dalam menanggapi fluktuasi pasar atau perubahan preferensi konsumen.

3.2.2.1.13 Linear Regression Visualisasi Model

Prediksi vs Aktual (Linear Regression)

R^2 Score (vs actual sale_price): 0.9316

RMSE (vs actual sale_price): 127.17

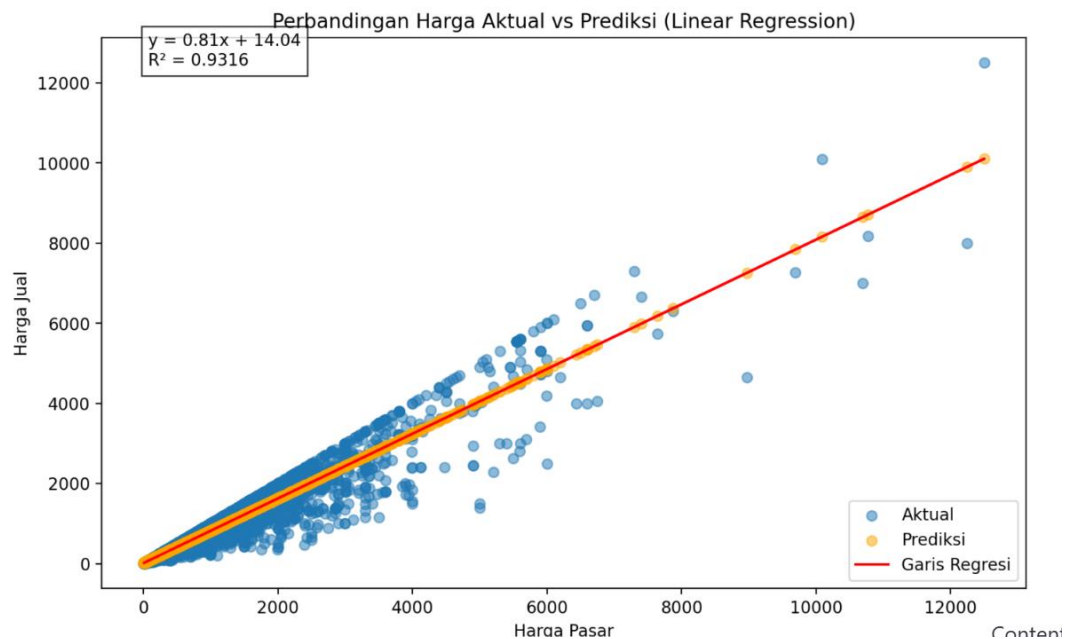


Figure 3.19 Tampilan Dashboard Koefisien Linear Regression

Grafik ini menampilkan koefisien yang dihasilkan oleh model regresi, yang menunjukkan pengaruh tiap variabel terhadap harga.

Dengan mengikuti garis prediksi bisa disimpulkan untuk penyebaran aktual tidak sejalan dengan garis prediksi yang terjadi pada rentang 40.000 hingga 12.000

3.2.2.2 Penjelasan Fitur Dashboard

- Mengupload file model (.pkl)



Figure 3.20 Memilih model (.pkl)

Memungkinkan pengguna untuk mengupload file model yang telah disimpan sebelumnya untuk diimplementasikan pada dashboard.

b. Menampilkan hasil

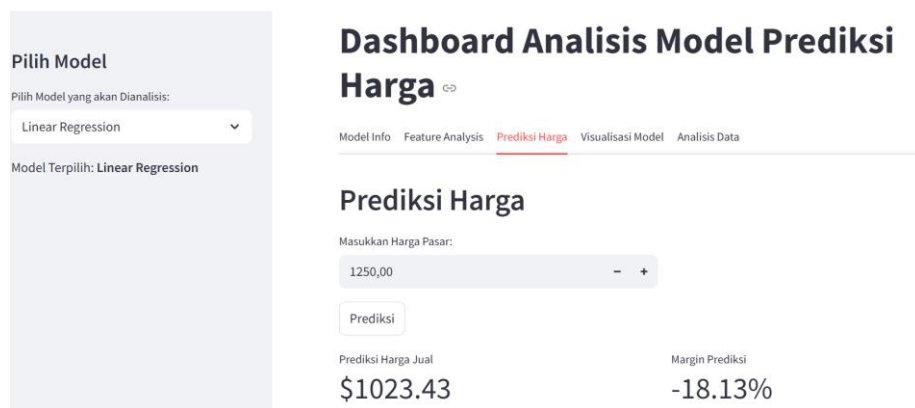


Figure 3.21 Menampilkan hasil prediksi harga Linear Regression

Visualisasi ini menampilkan harga yang diprediksi oleh model regresi linear.

Interpretasi:

Menampilkan hasil ini membantu memvalidasi model dan menyediakan estimasi harga yang dapat digunakan untuk benchmarking atau sebagai referensi untuk penyesuaian harga di masa depan.

Contoh Prediksi untuk Berbagai Harga

	Harga Pasar	Prediksi Harga Jual	Margin
0	\$100.00	\$94.79	-5.2%
1	\$500.00	\$417.80	-16.4%
2	\$1000.00	\$821.55	-17.8%
3	\$2000.00	\$1629.07	-18.5%
4	\$5000.00	\$4051.60	-19.0%

Figure 3.22 Menampilkan hasil prediksi harga Decision Tree

Membandingkan hasil ini dengan harga pasar saat ini bisa membantu menilai keakuratan dan relevansi model dalam kondisi pasar saat ini, memberikan dasar untuk penyesuaian atau perubahan model.

Informasi Model: Decision Tree

Tipe Model: Decision Tree Regressor

Kedalaman Pohon

8

Jumlah Leaf Nodes

206

Feature Importance

	Feature	Importance
0	market_price	1

Figure 3.23 Menampilkan hasil Feature Importance

Tabel ini menampilkan kepentingan relatif dari setiap fitur dalam model Decision Tree atau model regresi.

BAB 4

PENUTUP

4.1 Kesimpulan

Proyek ini berhasil mengatasi tantangan utama dalam analisis data produk, yaitu memprediksi harga jual dan mengelompokkan produk, melalui aplikasi teknik data mining. Model Decision Tree Regressor yang dikembangkan menunjukkan kemampuan yang memadai dalam memprediksi `sale_price` berdasarkan fitur `market_price` dari dataset. Analisis feature importance dari model Decision Tree mengkonfirmasi bahwa `market_price` adalah prediktor yang sangat berpengaruh, menjadikannya faktor kunci dalam penentuan harga jual.

Selain prediksi, metode K-Means Clustering berhasil mengelompokkan data produk berdasarkan karakteristik harga (`sale_price` dan `market_price`) ke dalam beberapa klaster yang berbeda. Pengelompokan ini memberikan wawasan berharga mengenai segmen produk dengan pola harga serupa, yang sebelumnya mungkin tidak terlihat secara eksplisit.

Salah satu luaran penting dari proyek ini adalah dashboard interaktif yang dibangun menggunakan Streamlit. Dashboard ini memungkinkan pengguna non-teknis untuk berinteraksi dengan hasil analisis: memuat dan memvisualisasikan model yang telah disimpan (linear regression, decision tree, kmeans menggunakan pickle), menjelajahi analisis fitur, dan melakukan prediksi harga baru secara real-time. Ini meningkatkan aksesibilitas temuan data mining.

Meskipun demikian, analisis ini terbatas oleh keragaman atribut yang tersedia dalam dataset. Keterbatasan pada fitur tambahan yang relevan menjadi kendala dalam menggali pola yang lebih kompleks atau meningkatkan akurasi prediksi dan kualitas pengelompokan lebih lanjut.

4.2 Saran

1. Strategi Penanganan Outlier dan Missing Value yang Lebih Lanjut: Meskipun penanganan dasar telah dilakukan, strategi yang lebih canggih untuk menangani outlier (misalnya, winsorizing, transformasi) dan missing value (misalnya, imputasi berbasis model atau KNN Imputer) dapat dieksplorasi untuk potensi peningkatan kualitas data

dan model.

2. Pertimbangan Analisis Deret Waktu: Jika data mencakup informasi temporal, analisis deret waktu dapat dilakukan untuk memahami tren harga atau pola penjualan dari waktu ke waktu, yang bisa menjadi masukan berharga untuk strategi penetapan harga dan inventaris.

DAFTAR PUSTAKA

Javatpoint. (n.d.). *K-Means Clustering Algorithm in Machine Learning*. Diakses pada 27 Desember 2024, dari <https://www.javatpoint.com/k-means-clustering-algorithm-in-machine-learning>

Medium. (2020, September 20). *Regresi Linear pada Machine Learning*. Diakses pada 27 Desember 2024, dari <https://medium.com/group-3-machine-learning/regresi-linear-pada-machine-learning-97f34ce18633>

Stack Overflow. (2019, December 13). *Decision Tree Too Big Scikit-Learn*. Diakses pada 27 Desember 2024, dari <https://stackoverflow.com/questions/59306728/decision-tree-too-big-scikit-learn>

Mohamad jajuli nurul rohmawati, sofi defiyanti, “Implementasi Algoritma K-Means Dalam Pengklasteran Mahasiswa Pelamar Beasiswa,” Jitter 2015, vol. I, no. 2, pp. 62–68, 2015.

LAMPIRAN

NIM	Nama	Pembagian Tugas
102022330283	Yunky Novfredly	BAB 2 KERANGKA KERJA: Modeling & Evaluation (Decision Tree, Mockup Dashboard). BAB 3 HASIL DAN PEMBAHASAN: Analisis hasil model (Linear Regression, Decision Tree, K-Means), Analisis Margin, Keuntungan & Keterbatasan Model, Penjelasan isi dashboard.
102022300266	Egi Agung Santoso Pardede	BAB 1 PENDAHULUAN: Latar Belakang, Rumusan Masalah, Tujuan, Manfaat, State of The Art (Metode Unsupervised: K-Means, Metode Supervised: Decision Tree & Regresi). BAB 4 PENUTUP: Kesimpulan dan Saran
102022300432	Fabert Varico	BAB 2 KERANGKA KERJA: Business Understanding, Sumber Data dan Dataset, Data Exploratory Analysis (Struktur Dataset, Tipe Data, Visualisasi Data), Data Preparation (Cleaning, Missing Value, Duplicated Data, Outlier). BAB 1 PENDAHULUAN: State of The Art (Metode Unsupervised: K-Means, Metode Supervised: Decision Tree & Regresi)

102022300181	Firdaus Yudha Sakti	BAB 2 KERANGKA KERJA (lanjutan): Modeling & Evaluation (K-Means Clustering, Linear Regression), Visualisasi Evaluasi Model (Elbow Method, Silhouette Score, Prediksi vs Aktual). BAB 4 PENUTUP: Kesimpulan dan Saran
--------------	----------------------------	--

Link Github : [Github](#)