**Mata Kuliah Aplikasi Data Scientist**

**Laporan Tugas P8: Linear Regresi**

**Dosen Pengampu: Ledy Elsera Astrianty, S.Kom., M.Kom.**



**Disusun oleh:**

* Lathif Ramadhan (5231811022)
* Andini Angel M. (5231811029)
* Rama Panji N. (5231811033)
* Giffari Riyanda P. (5231811036)

**PROGRAM STUDI SAINS DATA**

**FAKULTAS SAINS DAN TEKNOLOGI**

**UNIVERSITAS TEKNOLOGI YOGYAKARTA**

**YOGYAKARTA**

**2025**

# Daftar Isi

[Daftar Isi 2](#_Toc198535242)

[Bab 1: Pendahuluan 3](#_Toc198535243)

[Bab 2: Pembahasan 5](#_Toc198535244)

[2.1. Import Libraries 5](#_Toc198535245)

[2.2. Load Dataset 7](#_Toc198535246)

[2.3. Preprocessing (Penanganan Missing Value dan Duplikasi Data) 8](#_Toc198535247)

[2.3.1. Cek Duplikasi Data 8](#_Toc198535248)

[2.3.2. Cek Missing Value 9](#_Toc198535249)

[2.4. Exploratory Data Analysis (EDA) 10](#_Toc198535250)

[2.4.1. Deskripsi/Ringkasan Data 10](#_Toc198535251)

[2.4.2. Informasi Data (kolom) 10](#_Toc198535252)

[2.4.3. Distribusi Variabel Kategori 12](#_Toc198535253)

[2.4.4. Hubungan Variabel Kategorik dengan Target (charges) 14](#_Toc198535254)

[2.4.5. Distribusi Variabel Numerik 16](#_Toc198535255)

[2.4.6. Hubungan Variabel Numerik dengan Target (charges) 19](#_Toc198535256)

[2.4.6. Outlier Detection (Opsional) 20](#_Toc198535257)

[2.4.7. Analisis Korelasi Numerik 21](#_Toc198535258)

[2.5. Preprocessing (Persiapan untuk model Regresi) 22](#_Toc198535259)

[2.5.2. Konfirmasi Hasil Encoding 23](#_Toc198535260)

[2.5.3. Memisahkan fitur (X) dan target (y) 24](#_Toc198535261)

[2.5.4. Split data training-testing (80:20) 25](#_Toc198535262)

[2.5.5. Penskalaan Fitur Numerik 25](#_Toc198535263)

[2.5.6. Verifikasi Bentuk Data Setelah Split 27](#_Toc198535264)

[2.6. Pemodelan dan Simulasi 27](#_Toc198535265)

[2.6.1. Inisialisasi dan training model 27](#_Toc198535266)

[2.6.2. Prediksi data testing 28](#_Toc198535267)

[2.6.4. Analisis Residual 30](#_Toc198535268)

[2.7. Simulasi Prediksi 35](#_Toc198535269)

[2.7.1. Contoh 1: Non-perokok, laki-laki, 30 tahun, bmi 25, 1 anak, di southeast 35](#_Toc198535270)

[2.7.2. Contoh 2: Sama seperti contoh 1, tetapi perokok 35](#_Toc198535271)

[2.7.3. Perbedaan biaya karena merokok (untuk contoh ini) 36](#_Toc198535272)

[Bab 3: Kesimpulan 37](#_Toc198535273)

[Bab 4: Lampiran 39](#_Toc198535274)

# Bab 1: Pendahuluan

Dataset yang digunakan dalam proyek analisis ini merupakan data biaya asuransi kesehatan individu yang bersumber dari platform Kaggle (*Insurance Dataset*). Data ini awalnya dikumpulkan untuk mendukung pembelajaran dalam buku *"Machine Learning with R"* karya Brett Lantz. Meskipun dataset ini merupakan data publik, data tersebut telah melalui proses pembersihan dan penyesuaian format agar sesuai dengan kebutuhan analisis dalam buku tersebut. Proyek ini bertujuan untuk memanfaatkan dataset tersebut guna membangun model prediktif yang dapat memperkirakan biaya medis individu berdasarkan faktor-faktor demografis dan kebiasaan hidup.

**Link Sumber Dataset:** <https://www.kaggle.com/datasets/mirichoi0218/insurance>

**Deskripsi Variabel**  
Dataset terdiri dari **1.337 entri** (baris) dengan **7 variabel** (kolom) yang mencakup:

1. **Usia Tertanggung (age)**:  
   Menunjukkan usia penerima manfaat utama asuransi. Rentang usia dalam dataset ini berkisar antara **18 hingga 64 tahun**, dengan rata-rata usia **39.2 tahun** dan distribusi yang relatif merata (standar deviasi ±14 tahun).
2. **Jenis Kelamin (sex)**:  
   Kategori jenis kelamin tertanggung, yaitu **perempuan (female)** dan **laki-laki (male)**. Variabel ini termasuk dalam tipe data kategorik (*object*).
3. **Indeks Massa Tubuh (bmi)**:  
   Nilai BMI (*Body Mass Index*) yang dihitung berdasarkan rasio berat badan (kg) terhadap kuadrat tinggi badan (m²). Nilai BMI ideal umumnya berada di kisaran **18.5–24.9**, namun rata-rata BMI dalam dataset ini adalah **30.66**, mengindikasikan bahwa sebagian besar tertanggung memiliki berat badan di atas normal (*overweight* atau *obese*). Rentang BMI cukup lebar, mulai dari **15.96** (sangat kurus) hingga **53.13** (obesitas ekstrem).
4. **Jumlah Tanggungan (children)**:  
   Menunjukkan jumlah anak atau dependen yang tercakup dalam polis asuransi. Sebagian besar tertanggung memiliki **0–2 anak** (nilai median = 1), dengan maksimal **5 anak**.
5. **Status Merokok (smoker)**:  
   Variabel kategorik yang mengidentifikasi apakah tertanggung merupakan perokok (*yes*) atau bukan (*no*). Kebiasaan merokok diduga kuat berpengaruh signifikan terhadap biaya asuransi.
6. **Wilayah Tempat Tinggal (region)**:  
   Wilayah geografis tempat tinggal tertanggung di AS, terbagi menjadi empat kategori: *northeast*, *southeast*, *southwest*, dan *northwest*.
7. **Biaya Asuransi (charges)**:  
   Variabel target dalam proyek ini, yaitu biaya medis individu yang ditagihkan kepada perusahaan asuransi. Biaya ini memiliki variasi yang sangat besar, mulai dari **$1,121.87** hingga **$63,770.43**, dengan rata-rata **$13,279.12** dan standar deviasi tinggi (±$12,110.36). Sebanyak 25% tertanggung memiliki biaya di bawah **$4,746**, sementara 75% di bawah **$16,657**.

**Kualitas Data**  
Berdasarkan pemeriksaan menggunakan df.info(), seluruh kolom memiliki **1.337 entri** tanpa nilai kosong (*non-null*), yang mengonfirmasi bahwa data sudah bersih dari *missing values*. Tipe data untuk setiap kolom juga sudah sesuai:

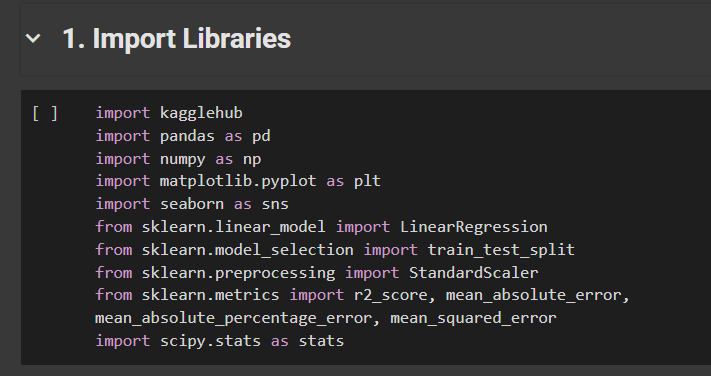
* Numerik (*float64* dan *int64*): age, bmi, children, charges
* Kategorik (*object*): sex, smoker, region

**Relevansi dengan Proyek**  
Dataset ini dipilih karena memuat variabel-variabel kunci yang secara logis berkaitan dengan biaya medis, seperti usia, kebiasaan merokok, dan kondisi kesehatan (BMI). Variasi yang tinggi pada biaya asuransi (charges) menunjukkan kompleksitas hubungan antar variabel, sehingga analisis ini akan fokus pada identifikasi pola dan faktor dominan yang memengaruhi biaya. Hasil akhir dari proyek ini diharapkan dapat memberikan insight bagi perusahaan asuransi dalam memperkirakan risiko klaim dan menyusun strategi penetapan premi yang lebih akurat.

Berikut link file .ipynb pengerjaan proyek untuk kasus ini:

# Bab 2: Pembahasan

## 2.1. Import Libraries



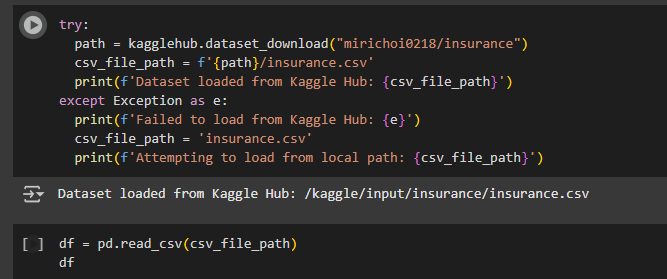
Bagian awal dari skrip analisis ini adalah serangkaian perintah import. Dalam konteks pemrograman Python, import berfungsi untuk memuat (atau *'mengimpor'*) pustaka dan modul eksternal ke dalam lingkungan kerja saat ini. Pustaka dan modul ini berisi kumpulan fungsi, kelas, dan alat yang telah dibuat sebelumnya, yang sangat membantu dalam melakukan tugas-tugas analisis data dan pemodelan tanpa harus membangun semuanya dari awal.

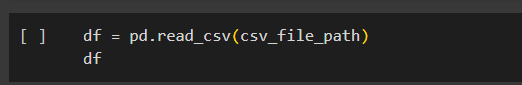
Berikut adalah penjelasan setiap baris impor, dikaitkan dengan perannya dalam proyek prediksi biaya asuransi ini:

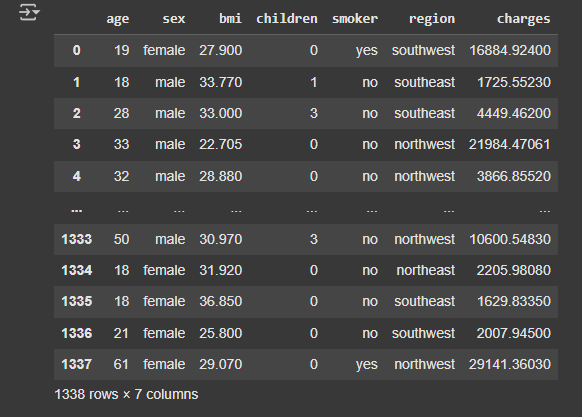
* import kagglehub: Pustaka ini diimpor untuk memfasilitasi pengambilan data. Dalam proyek ini, kagglehub digunakan untuk mengunduh dataset *'insurance.csv'* yang bersumber dari Kaggle secara langsung ke lingkungan komputasi. Hal ini memungkinkan akses mudah ke data yang akan dianalisis.
* import pandas as pd: pandas adalah pustaka fundamental untuk manipulasi dan analisis data dalam bentuk tabel (*DataFrame*). Setelah dataset berhasil diunduh, pandas digunakan untuk membacanya menjadi DataFrame, memungkinkan operasi seperti melihat struktur data, memeriksa baris duplikat, menangani nilai yang hilang (walaupun pada dataset ini tidak ada), serta melakukan transformasi data dan *encoding* variabel kategorik. Alias pd diberikan untuk efisiensi dalam penulisan kode.
* import numpy as np: numpy menyediakan dukungan untuk *array* dan matriks multi-dimensi, serta berbagai fungsi matematika tingkat tinggi yang beroperasi pada struktur data tersebut. Dalam proyek ini, numpy diperlukan untuk perhitungan numerik, seperti menghitung akar kuadrat saat menghitung *Root Mean Squared Error* (RMSE), dan juga menjadi basis bagi banyak operasi komputasi dalam pustaka lain. Alias np diberikan.
* import matplotlib.pyplot as plt: matplotlib adalah pustaka dasar untuk membuat visualisasi statis dan interaktif di Python. Sub-modul pyplot menyediakan antarmuka yang sederhana untuk membuat berbagai jenis plot, seperti histogram, boxplot, dan scatter plot. Dalam tahap *Exploratory Data Analysis* (EDA) proyek ini, matplotlib.pyplot digunakan untuk memvisualisasikan distribusi variabel dan hubungan antar variabel untuk mendapatkan pemahaman awal tentang data asuransi. Alias plt diberikan.
* import seaborn as sns: seaborn adalah pustaka visualisasi data yang dibangun di atas matplotlib. seaborn menyediakan fungsi-fungsi untuk membuat grafik statistik yang lebih kompleks dan menarik dengan sedikit kode. Di proyek ini, seaborn digunakan untuk menghasilkan visualisasi seperti distribusi dengan kurva densitas, boxplot untuk membandingkan grup, scatter plot dengan pemisah kategori, dan heatmap korelasi, yang semuanya memperkaya analisis visual data asuransi. Alias sns diberikan.
* from sklearn.linear\_model import LinearRegression: Scikit-learn (sklearn) adalah pustaka terdepan untuk *machine learning* di Python. Dari modul linear\_model, kelas LinearRegression diimpor. Kelas ini mengimplementasikan model regresi linier, yang merupakan algoritma utama yang dipilih dalam proyek ini untuk memprediksi biaya asuransi (charges) sebagai fungsi linier dari fitur-fitur input.
* from sklearn.model\_selection import train\_test\_split: Dari modul model\_selection Scikit-learn, fungsi train\_test\_split diimpor. Fungsi ini krusial untuk membagi dataset secara acak menjadi dua subset: data pelatihan (*training set*) dan data pengujian (*test set*). Data pelatihan digunakan untuk melatih model, sementara data pengujian digunakan untuk mengevaluasi kinerja model pada data yang belum pernah dilihat sebelumnya, memastikan evaluasi yang objektif.
* from sklearn.preprocessing import StandardScaler: Dari modul preprocessing Scikit-learn, kelas StandardScaler diimpor. Kelas ini digunakan untuk menerapkan teknik penskalaan standar (*standardization*) pada fitur-fitur numerik (seperti usia dan BMI). Penskalaan ini mentransformasi data sehingga memiliki rata-rata nol dan deviasi standar satu, sebuah langkah penting dalam pra-pemrosesan data untuk model regresi linier agar semua fitur memberikan kontribusi yang setara tanpa dipengaruhi oleh skala aslinya.
* from sklearn.metrics import r2\_score, mean\_absolute\_error, mean\_absolute\_percentage\_error, mean\_squared\_error: Dari modul metrics Scikit-learn, berbagai fungsi untuk evaluasi performa model diimpor. Fungsi-fungsi ini meliputi:
  + r2\_score: Untuk menghitung koefisien determinasi (R-squared), yang mengukur proporsi variabilitas variabel target yang dapat dijelaskan oleh model.
  + mean\_absolute\_error (MAE): Untuk menghitung rata-rata selisih absolut antara nilai prediksi dan nilai aktual.
  + mean\_absolute\_percentage\_error (MAPE): Untuk menghitung rata-rata persentase selisih absolut, memberikan interpretasi kesalahan dalam persentase.
  + mean\_squared\_error (MSE): Untuk menghitung rata-rata kuadrat selisih, memberikan penalti yang lebih besar untuk kesalahan yang besar. Metrik-metrik ini digunakan untuk mengukur seberapa baik model regresi linier yang dibangun dalam memprediksi biaya asuransi pada data pengujian.
* import scipy.stats as stats: Dari pustaka scipy (SciPy, pustaka untuk komputasi ilmiah dan teknis), modul stats diimpor. Modul ini berisi berbagai alat statistik dan fungsi distribusi probabilitas. Dalam analisis residual proyek ini, scipy.stats digunakan untuk membuat Q-Q plot (*Quantile-Quantile plot*) dari residual, yang merupakan metode visual untuk memeriksa apakah residual mengikuti distribusi normal, salah satu asumsi kunci dalam regresi linier. Alias stats diberikan.

Dengan mengimpor pustaka-pustaka ini di awal skrip, semua alat dan fungsi yang diperlukan untuk setiap tahapan proyek — mulai dari pengambilan data, pra-pemrosesan, analisis eksplorasi, pembangunan model, hingga evaluasi kinerja model — menjadi tersedia dan siap digunakan.

## 2.2. Load Dataset

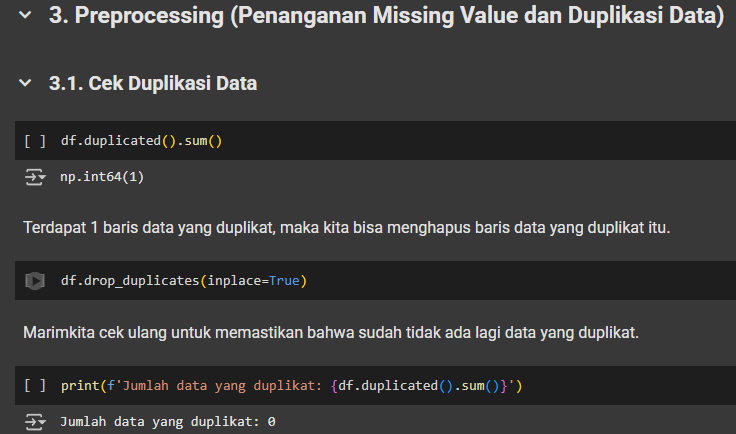




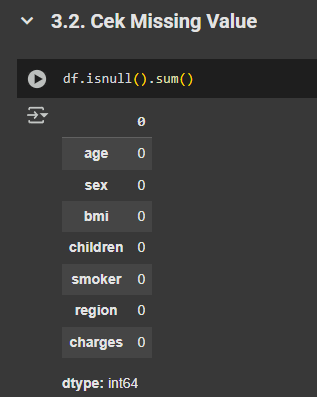


## 2.3. Preprocessing (Penanganan Missing Value dan Duplikasi Data)

### 2.3.1. Cek Duplikasi Data



### 2.3.2. Cek Missing Value



Setiap kolom menampilkan angka 0: Artinya tidak ada missing value di seluruh kolom (age, sex, bmi, dll.).

Mengapa harus mengecek missing value pada data ini?

1. Modeling/Regresi

Algoritma machine learning seperti Linear Regression tidak bisa bekerja dengan missing value.

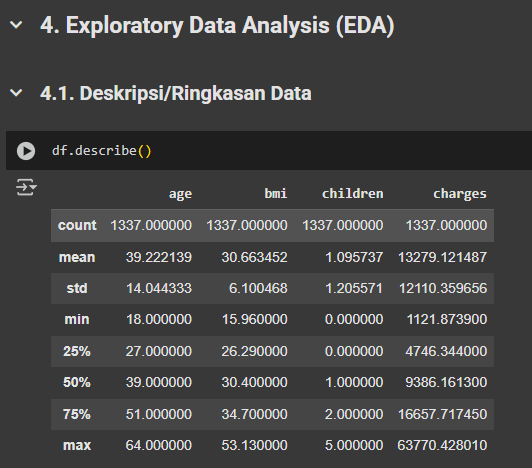
Jika ada NaN, perlu di-handle sebelum training model.

2. Analisis Statistik

Missing value dapat mengganggu perhitungan mean, median, atau korelasi.

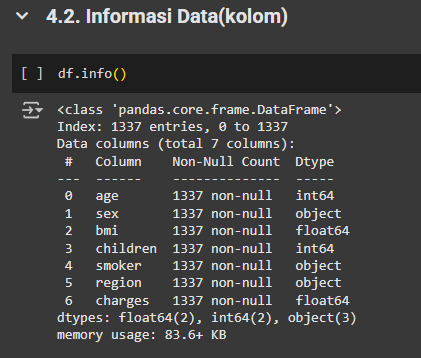
## 2.4. Exploratory Data Analysis (EDA)

### 2.4.1. Deskripsi/Ringkasan Data



Pernyataan df.describe() adalah metode dalam pustaka pandas yang secara otomatis menghitung dan menampilkan ringkasan statistik deskriptif untuk semua kolom yang berisi data numerik dalam DataFrame df. Output dari perintah ini memberikan gambaran cepat mengenai distribusi nilai dalam setiap kolom numerik, yang sangat penting di awal tahap Exploratory Data Analysis (EDA).

### 2.4.2. Informasi Data (kolom)



Output df.info() dan Relevansinya dalam Proyek Prediksi Biaya Asuransi

Output dari df.info() memberikan informasi penting yang langsung relevan dengan langkah-langkah selanjutnya dalam proyek prediksi biaya asuransi ini:

* <class 'pandas.core.frame.DataFrame'>: Ini mengkonfirmasi bahwa objek df memang adalah sebuah DataFrame pandas.
* Index: 1337 entries, 0 to 1337: Menunjukkan jumlah baris data dalam DataFrame ini setelah duplikasi dihapus (Kita sebelumnya menghapus 1 duplikat dari 1338 baris asli, sehingga tersisa 1337 baris). Ini adalah jumlah sampel individu dalam dataset yang akan kita gunakan. Rentang indeksnya dari 0 hingga 1337.
* Data columns (total 7 columns):: Ini memberi tahu kita bahwa DataFrame ini memiliki total 7 kolom (atau variabel).

Bagian tabel di bawahnya memberikan detail per kolom:

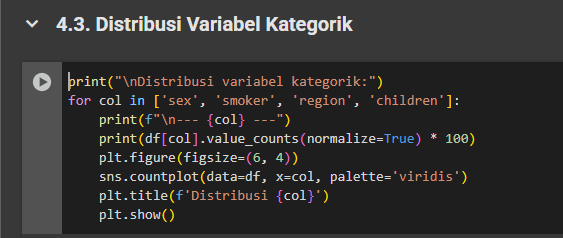
1. #: Nomor indeks kolom.
2. Column: Nama kolom (misalnya, age, sex, bmi, dll.). Ini adalah nama-nama fitur dan target yang akan kita gunakan.
3. Non-Null Count: Menunjukkan jumlah nilai yang tidak kosong (non-missing) di setiap kolom. Untuk semua kolom (age hingga charges), nilainya adalah 1337, yang sama dengan jumlah total baris. Ini sangat penting karena mengkonfirmasi temuan kita sebelumnya bahwa tidak ada nilai yang hilang (missing values) dalam dataset ini. Ini berarti kita tidak perlu melakukan imputasi atau penghapusan baris/kolom karena missing value.
4. Dtype: Menunjukkan tipe data dari nilai-nilai dalam kolom tersebut.

* int64: Bilangan bulat (seperti age dan children). Ini adalah tipe data numerik diskrit.
* float64: Bilangan riil/desimal (seperti bmi dan charges). Ini adalah tipe data numerik kontinu. Variabel target kita, charges, bertipe float, sesuai untuk masalah regresi.
* object: Umumnya menandakan string atau campuran tipe data. Dalam kasus ini, ini menunjukkan kolom-kolom kategorik (sex, smoker, region) yang berisi teks.
* dtypes: float64(2), int64(2), object(3): Ini adalah ringkasan jumlah kolom berdasarkan tipe datanya. Ada 2 kolom float, 2 kolom integer, dan 3 kolom bertipe object. Informasi tipe data ini krusial karena memengaruhi cara kita memproses kolom-kolom ini. Kolom numerik (int64, float64) bisa langsung digunakan dalam perhitungan atau diskalakan (StandardScaler), sementara kolom object (kategorik) perlu di-encoding (seperti yang kita lakukan dengan pd.get\_dummies) sebelum dimasukkan ke dalam model regresi yang membutuhkan input numerik.
* memory usage: 83.6+ KB: Memberikan perkiraan penggunaan memori oleh DataFrame.

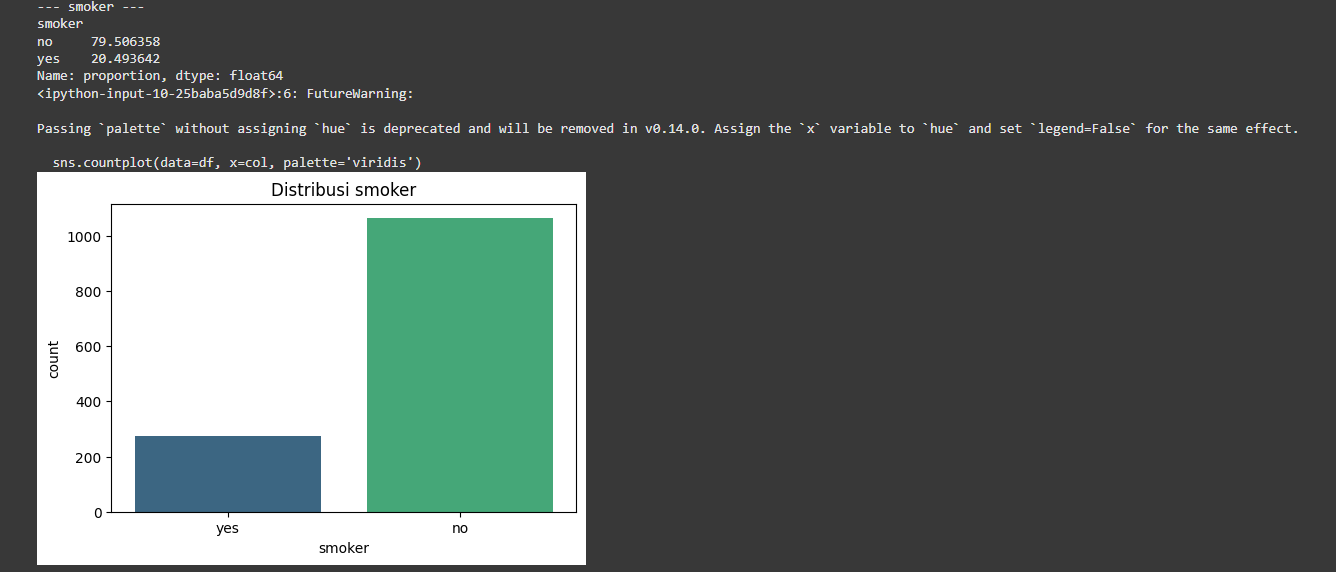
Singkatnya, output df.info() adalah \*"kartu identitas"\* dataset ini. Dalam proyek ini, ini memberikan konfirmasi vital bahwa data ini:

1. Bersih dari missing values
2. Dengan jelas mengidentifikasi mana kolom numerik (siap untuk perhitungan/penskalaan) dan mana kolom kategorik (membutuhkan encoding)
3. Membimbing langkah-langkah pra-pemrosesan data ini selanjutnya

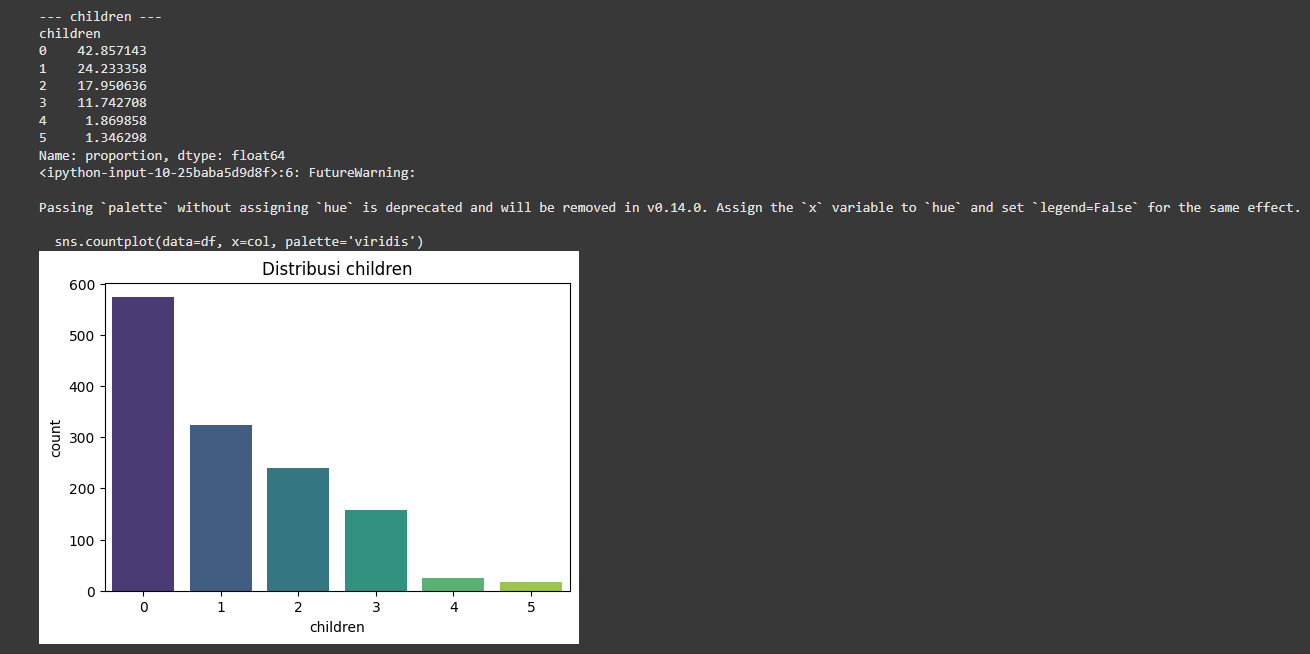
### 2.4.3. Distribusi Variabel Kategori



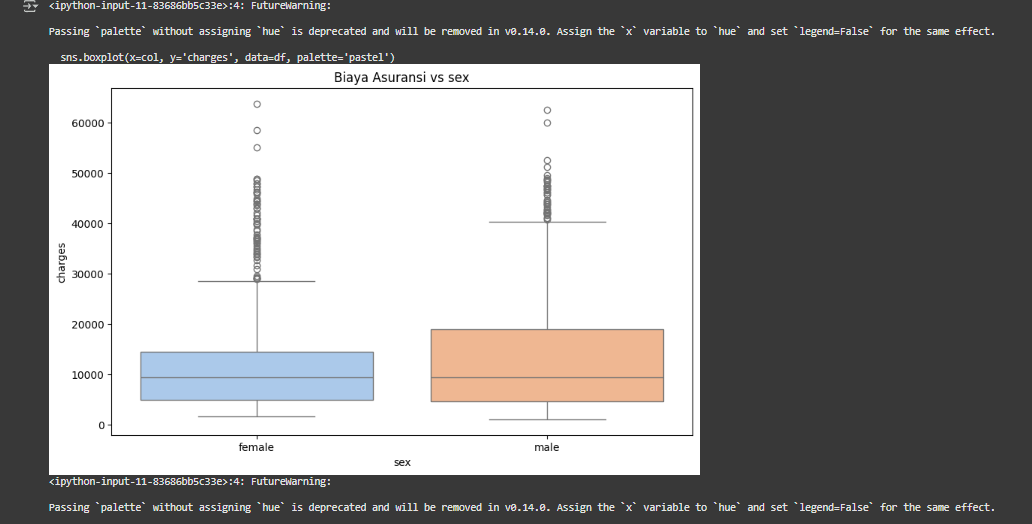
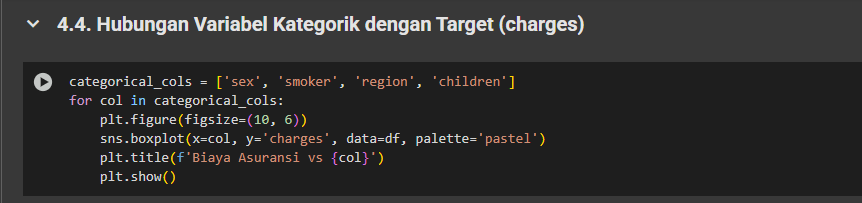


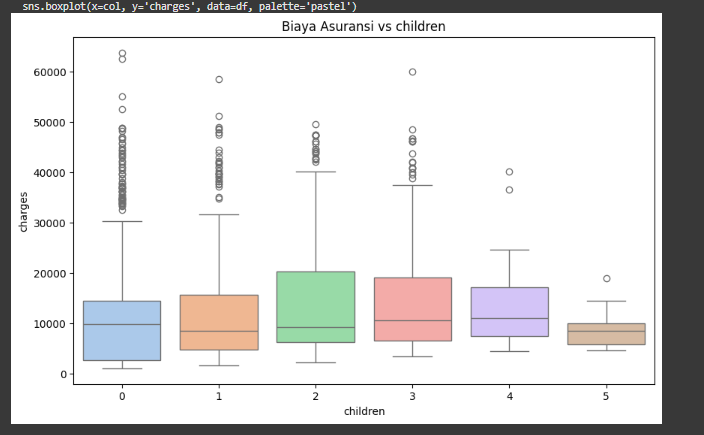
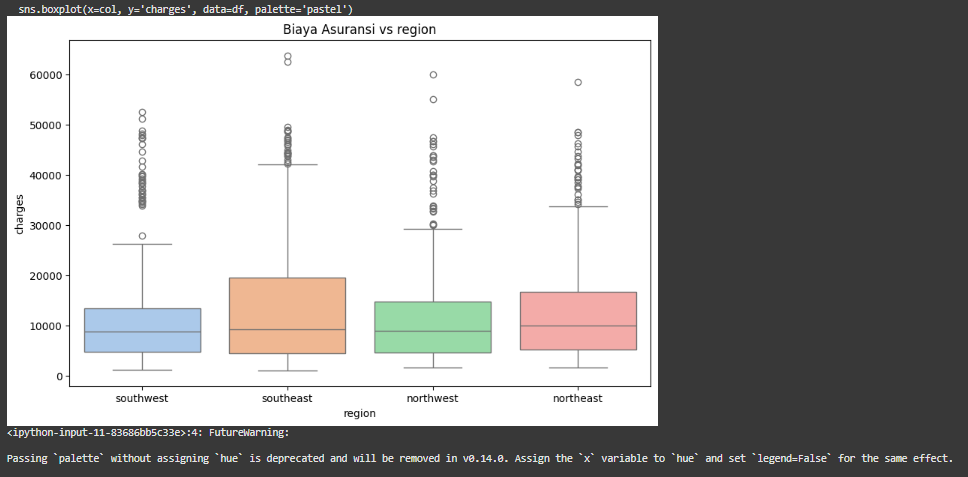
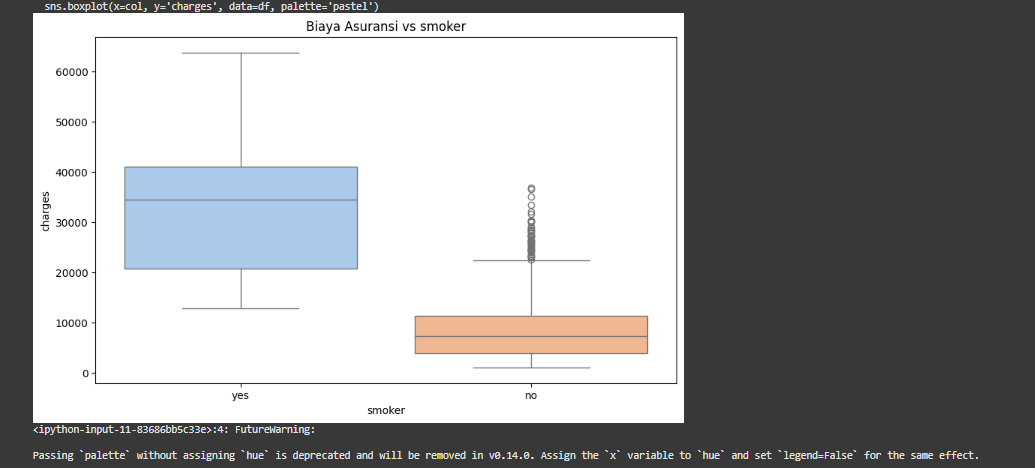




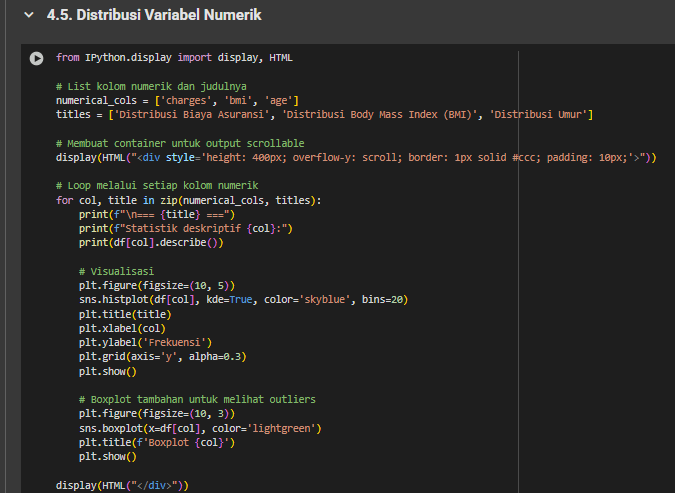


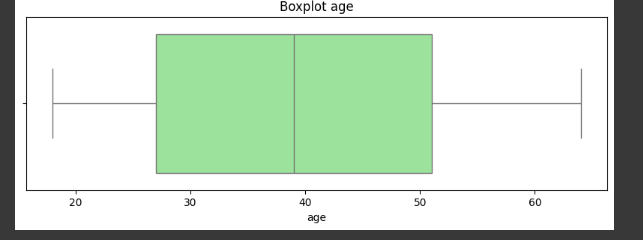
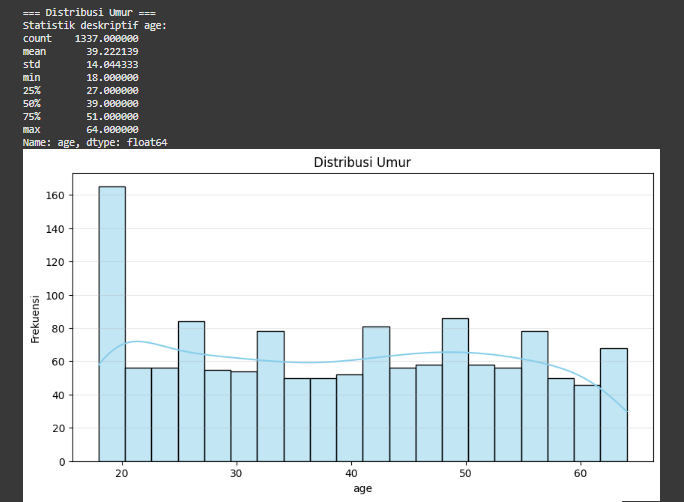
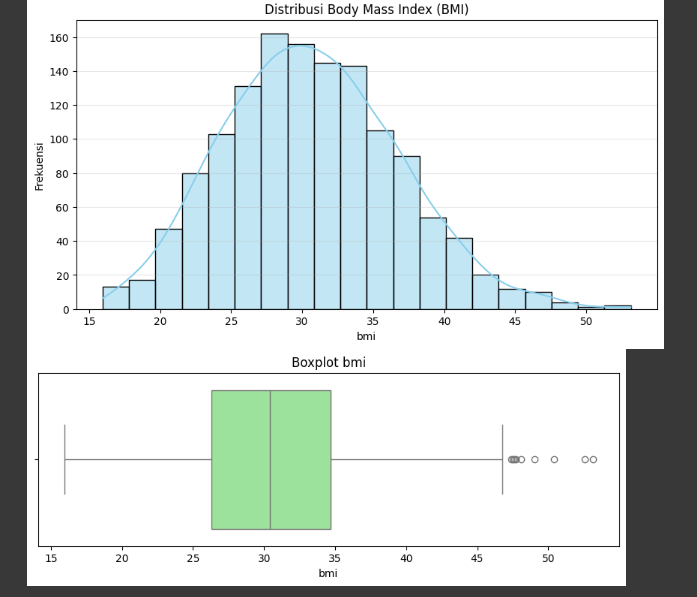
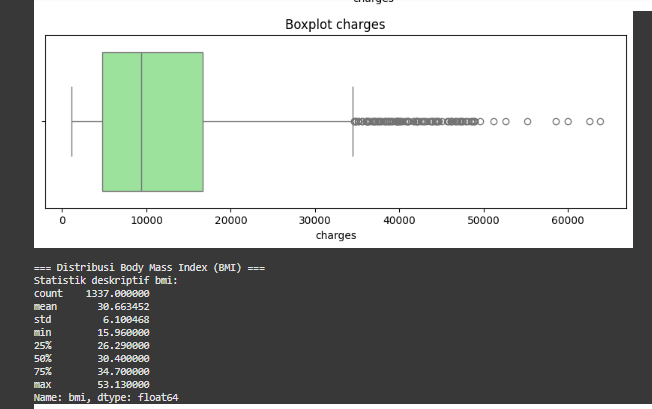
### 2.4.4. Hubungan Variabel Kategorik dengan Target (charges)



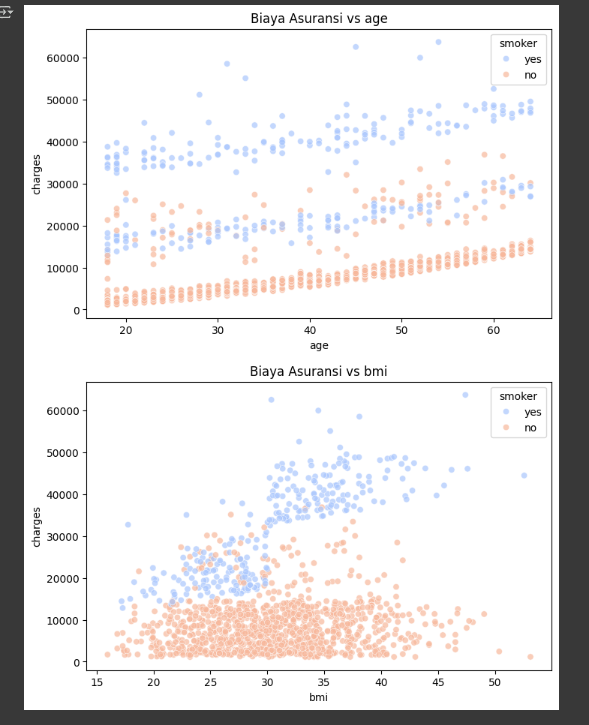
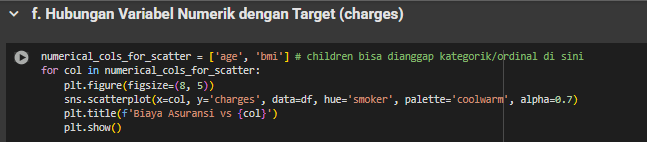


### 2.4.5. Distribusi Variabel Numerik

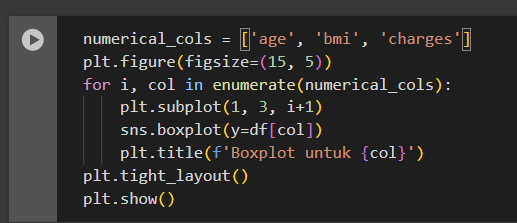


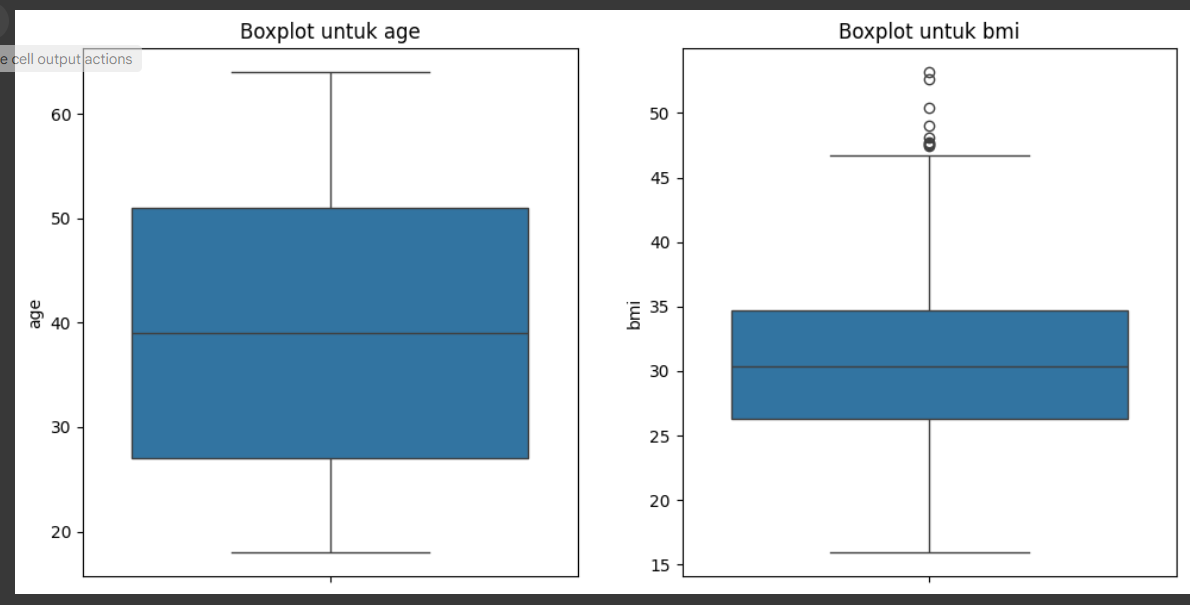


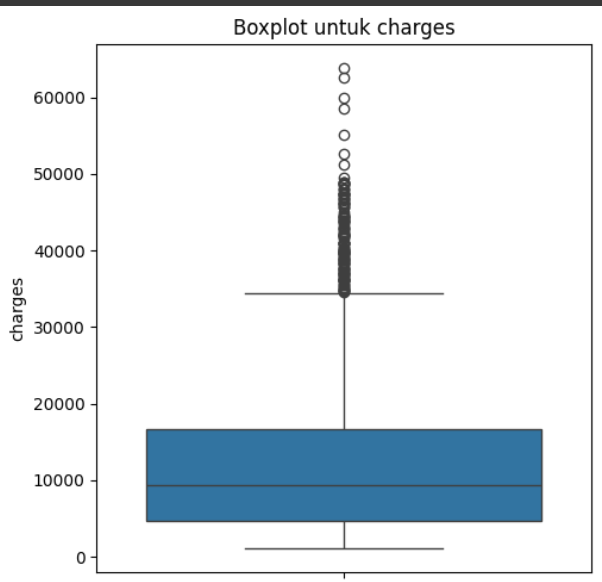
### 2.4.6. Hubungan Variabel Numerik dengan Target (charges)



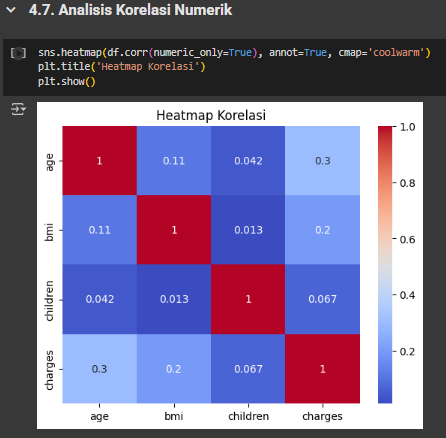
### 2.4.6. Outlier Detection (Opsional)







### 2.4.7. Analisis Korelasi Numerik



Berikut penjelasan hasil analisis korelasi antara variabel-variabel numerik dalam dataset asuransi kesehatan:

**1. Korelasi Umum Antar Variabel**

* Usia (age) dengan Biaya (charges):

Menunjukkan korelasi positif sedang (0.3). Artinya, semakin tua usia tertanggung, cenderung memiliki biaya asuransi yang lebih tinggi. Hal ini masuk akal karena risiko kesehatan umumnya meningkat seiring usia.

* BMI dengan Biaya (charges):

Korelasi positif lemah (0.2). Nilai BMI yang lebih tinggi sedikit berkaitan dengan biaya asuransi yang lebih besar, meskipun pengaruhnya tidak terlalu kuat.

* Jumlah Anak (children) dengan Variabel Lain:
  + Korelasi sangat lemah dengan usia (0.042) dan BMI (0.013), menunjukkan hampir tidak ada hubungan.
  + Korelasi negatif lemah dengan biaya (-0.2), artinya jumlah anak tidak terlalu memengaruhi biaya asuransi dalam dataset ini.

**2. Pola Menarik**

* Korelasi Terkuat:

Usia dengan biaya asuransi (0.3) adalah korelasi tertinggi, menjadikan usia sebagai prediktor potensial yang penting dalam model prediksi biaya.

* BMI:

Meski memiliki korelasi lemah dengan biaya (0.2), kombinasi dengan faktor lain (seperti status merokok) mungkin berpengaruh lebih signifikan.

* Anak (children):

Korelasi negatif dengan biaya (-0.2) mungkin mengejutkan, tetapi nilai yang kecil menunjukkan pengaruhnya minimal. Perlu investigasi lebih lanjut apakah ada faktor lain yang berperan.

**3. Implikasi untuk Proyek Ini**

* Variabel Prediktor Utama:

Usia layak dipertimbangkan sebagai fitur utama dalam model regresi karena korelasinya yang signifikan dengan biaya.

* Transformasi Data:

Kolom seperti BMI mungkin perlu ditransformasi (contoh: dikelompokkan dalam kategori) untuk meningkatkan korelasi jika digunakan dalam model.

* Variabel Kategorikal:

Variabel seperti smoker (tidak termasuk dalam heatmap karena non-numerik) mungkin memiliki pengaruh besar. Pastikan untuk meng-encode variabel ini dalam pra-pemrosesan.

* Multikolinearitas:

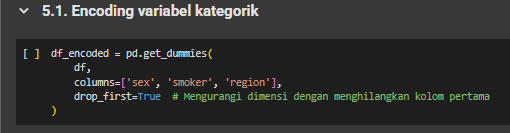
Tidak ada korelasi tinggi antar variabel independen (misal: age dan BMI hanya 0.11), sehingga risiko multikolinearitas rendah.

**Catatan Penting**

Nilai korelasi di atas berdasarkan data mentah. Setelah pra-pemrosesan (encoding variabel kategorikal, penskalaan), pola korelasi mungkin berubah.

Korelasi tidak berarti sebab-akibat. Faktor lain (seperti status merokok atau kondisi medis) mungkin menjadi penyebab sebenarnya dari biaya tinggi.

## 2.5. Preprocessing (Persiapan untuk model Regresi)

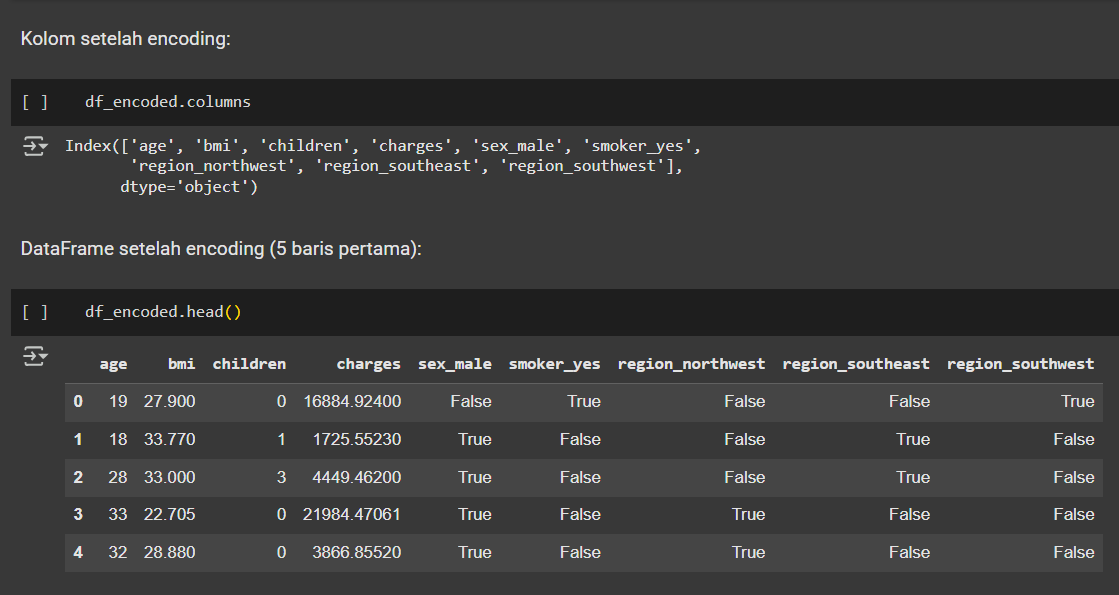


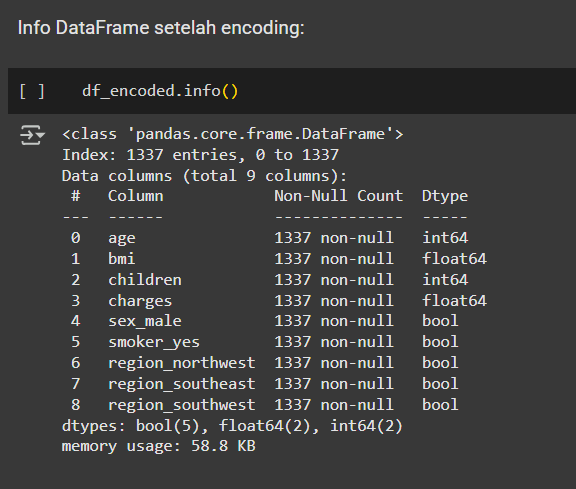
Proses One-Hot Encoding ini sangat penting dalam proyek prediksi biaya asuransi ini karena:

* Menyiapkan Data untuk Model: Model LinearRegression di Scikit-learn memerlukan semua input fitur dalam format numerik. One-Hot Encoding mengubah variabel kategorik menjadi format numerik (biner 0/1) yang dapat diproses oleh model.
* Mempertahankan Informasi: Teknik ini memastikan bahwa informasi yang terkandung dalam variabel kategorik (yaitu, individu termasuk dalam kategori tertentu atau tidak) tetap ada dan dapat digunakan oleh model untuk membuat prediksi.

### 2.5.2. Konfirmasi Hasil Encoding

Bagian ini terdiri dari tiga baris kode terpisah yang masing-masing berfungsi untuk menampilkan informasi berbeda mengenai DataFrame df\_encoded yang baru saja dibuat melalui One-Hot Encoding. Tujuannya adalah untuk memastikan bahwa kolom-kolom kategorik asli telah diganti dengan kolom-kolom biner yang dihasilkan oleh pd.get\_dummies(), dan bahwa struktur DataFrame sudah sesuai.



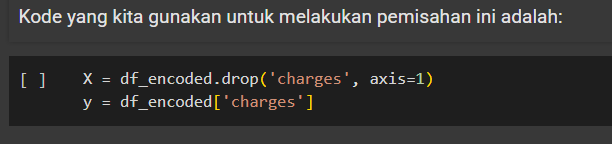


Secara keseluruhan, ketiga baris kode ini bersama-sama berfungsi sebagai pemeriksaan kualitas setelah One-Hot Encoding, memastikan bahwa data kita telah berhasil ditransformasikan ke format yang siap untuk digunakan dalam pembangunan model prediksi biaya asuransi.

### 2.5.3. Memisahkan fitur (X) dan target (y)

Dalam pembangunan model machine learning, terutama untuk masalah regresi seperti prediksi biaya asuransi, dataset perlu dibagi menjadi dua bagian utama:

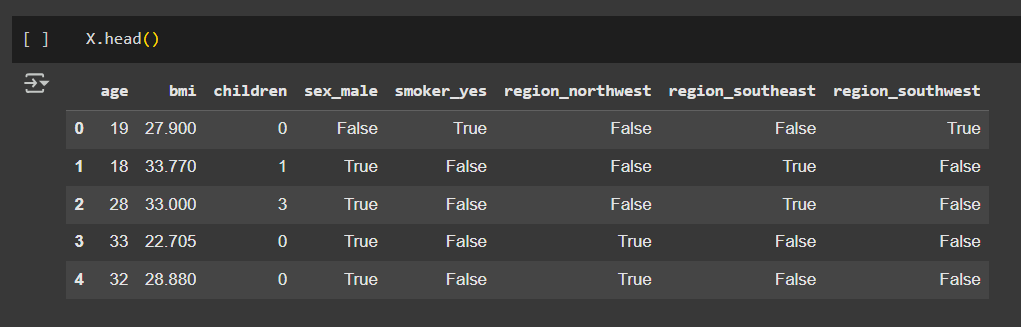
1. Fitur (Features), sering dilambangkan dengan X: Ini adalah kolom-kolom input yang akan digunakan model untuk "belajar" dan membuat prediksi. Fitur-fitur ini adalah karakteristik dari setiap individu (usia, BMI, jenis kelamin, status perokok, wilayah, jumlah anak, dll.).
2. Target (Target Variable atau Label), sering dilambangkan dengan y: Ini adalah kolom output yang ingin diprediksi oleh model. Dalam proyek ini, targetnya adalah biaya asuransi (charges).



Fitur (X) - 5 baris pertama:

Kita akan melihat tabel yang menampilkan 5 baris pertama dari DataFrame X. Tabel ini akan berisi semua kolom numerik dan biner yang sebelumnya ada di df\_encoded, kecuali kolom charges. Anda akan melihat kolom seperti age, bmi, children, sex\_male, smoker\_yes, dan kolom region\_....

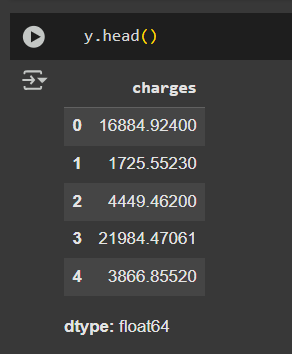
Ini mengkonfirmasi bahwa X berhasil dibuat dan hanya berisi fitur-fitur input yang akan digunakan model untuk memprediksi.



Target (y) - 5 baris pertama:

Kita akan melihat output berupa Series (bukan tabel) yang menampilkan 5 nilai pertama dari variabel target charges. Output ini hanya akan berisi angka-angka yang merepresentasikan biaya asuransi.

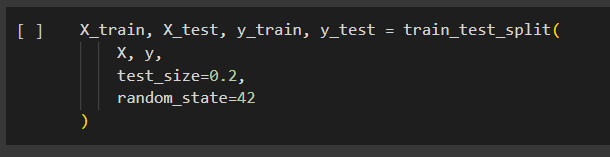
Ini mengkonfirmasi bahwa y berhasil dibuat dan hanya berisi variabel yang akan menjadi target prediksi model.



Dengan melakukan pemisahan ini, kita secara eksplisit mendefinisikan mana data yang akan menjadi masukan (fitur) dan mana data yang akan menjadi keluaran (target) untuk model regresi linear kita. Ini adalah langkah fundamental sebelum membagi data menjadi set pelatihan dan pengujian, serta melatih model.

### 2.5.4. Split data training-testing (80:20)

Setelah data pra-pemrosesan (seperti One-Hot Encoding dan pemisahan fitur/target) selesai, langkah krusial sebelum melatih model adalah membagi dataset yang ada menjadi dua bagian terpisah: satu bagian untuk melatih model (training set) dan satu bagian lainnya untuk menguji seberapa baik model tersebut bekerja pada data yang belum pernah dilihat sebelumnya (test set). Tujuan pembagian ini adalah untuk mendapatkan evaluasi model yang objektif dan menghindari overfitting (di mana model terlalu "menghafal" data pelatihan tetapi buruk dalam memprediksi data baru).



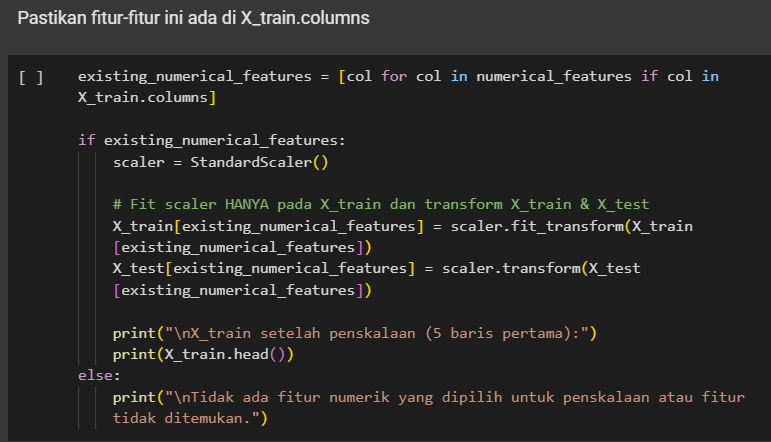
Dengan membagi data seperti ini, kita memitigasi risiko overfitting dan memastikan bahwa metrik evaluasi performa model ini benar-benar mencerminkan kemampuan generalisasi model pada data yang tidak digunakan selama pelatihan.

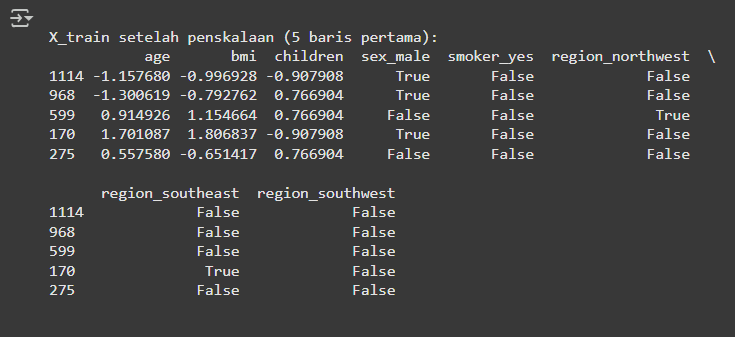
### 2.5.5. Penskalaan Fitur Numerik

Sebelum melatih model regresi linier, seringkali disarankan untuk melakukan penskalaan (scaling) pada fitur-fitur numerik. Fitur-fitur seperti usia (age), BMI (bmi), dan jumlah anak (children) memiliki rentang nilai yang berbeda-beda. Misalnya, usia bisa berkisar dari belasan hingga puluhan, sementara BMI mungkin berkisar dari 15 hingga 50, dan jumlah anak dari 0 hingga 5. Model regresi linier, meskipun tidak sesensitif model berbasis jarak, terkadang dapat sedikit ditingkatkan performanya jika fitur-fitur numerik berada dalam skala yang serupa. Selain itu, standardisasi ini merupakan praktik umum dan penting untuk model lain di Scikit-learn.

Metode penskalaan yang umum digunakan adalah Standardization (Standard Scaler). Proses ini mengubah nilai setiap fitur sehingga rata-ratanya menjadi 0 dan standar deviasinya menjadi 1.

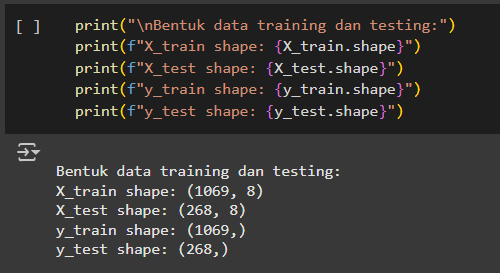






### 2.5.6. Verifikasi Bentuk Data Setelah Split

Setelah membagi dataset fitur (X) dan target (y) menjadi set pelatihan (X\_train, y\_train) dan set pengujian (X\_test, y\_test) menggunakan fungsi train\_test\_split(), langkah penting berikutnya adalah memverifikasi apakah pembagian tersebut menghasilkan jumlah baris dan kolom yang sesuai dengan proporsi yang diinginkan. Memastikan dimensi data (bentuk atau shape) ini sudah benar sangat penting sebelum melatih model, karena dimensi yang salah dapat menyebabkan error saat proses pelatihan.



## 2.6. Pemodelan dan Simulasi

### 2.6.1. Inisialisasi dan training model

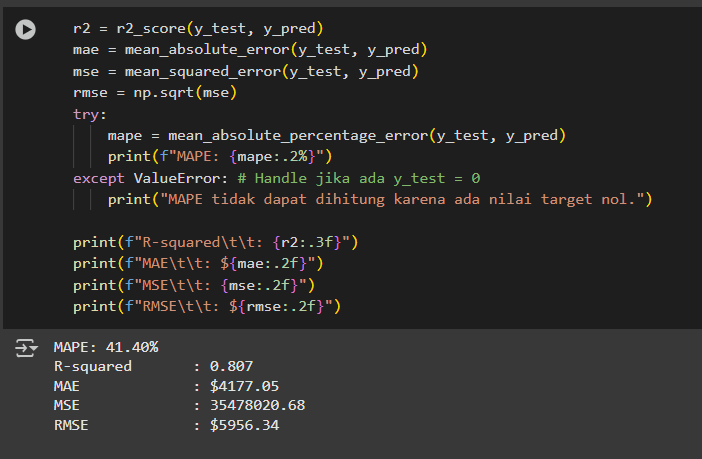
Setelah dataset siap — sudah dibersihkan, di-encode, diskalakan (fitur numerik), dan dibagi menjadi set pelatihan (X\_train, y\_train) serta set pengujian (X\_test, y\_test) — langkah berikutnya adalah membangun model prediksi. Dalam proyek ini, kita memilih model Regresi Linier untuk memprediksi biaya asuransi (charges).



Setelah baris kode ini dieksekusi, objek model kini berisi semua informasi yang dibutuhkan untuk melakukan prediksi pada data baru. Model ini sekarang memiliki koefisien yang spesifik untuk setiap fitur (age, bmi, children, sex\_male, smoker\_yes, region\_...) dan sebuah nilai intercept yang ditemukan selama proses pelatihan. Model ini siap untuk digunakan pada tahap selanjutnya, yaitu membuat prediksi pada data pengujian (X\_test).

### 2.6.2. Prediksi data testing

Setelah model Regresi Linier kita berhasil dilatih menggunakan set data pelatihan (X\_train dan y\_train), langkah berikutnya adalah menggunakan model yang sudah "belajar" tersebut untuk membuat prediksi. Proses prediksi ini dilakukan pada set data pengujian (X\_test), yaitu data yang belum pernah dilihat oleh model selama proses pelatihan.



**Hasil Evaluasi Model Regresi Linier**

1. **MAPE (Mean Absolute Percentage Error): 41.40%**
   * Secara rata-rata, prediksi biaya asuransi meleset sekitar **41.40%** dari nilai sebenarnya pada data pengujian.
   * Angka ini menunjukkan bahwa meskipun model sudah belajar dari data, masih ada variabilitas signifikan dalam biaya asuransi yang belum sepenuhnya dapat dijelaskan oleh fitur-fitur dalam model. Hal ini mengindikasikan ruang untuk perbaikan, mungkin dengan:
     + Menambahkan fitur lain
     + Mencoba model yang lebih kompleks
     + Memproses outlier atau nilai ekstrem
2. **R-squared (Koefisien Determinasi): 0.807**
   * Nilai **0.807** berarti sekitar **80.7%** variasi dalam biaya asuransi dapat dijelaskan oleh fitur-fitur yang digunakan.
   * Nilai ini dianggap cukup baik untuk data dunia nyata, menunjukkan bahwa fitur-fitur yang dipilih (*age*, *bmi*, *children*, dan variabel hasil encoding) merupakan prediktor yang relevan.
3. **MAE (Mean Absolute Error): $4,177.05**
   * Rata-rata selisih absolut antara prediksi dan nilai aktual adalah **$4,177.05**.
   * Memberikan ukuran kesalahan dalam satuan dolar yang mudah dipahami, menunjukkan akurasi model dari sudut pandang rata-rata kesalahan.
4. **MSE (Mean Squared Error): 35,478,020.68**
   * Rata-rata kuadrat selisih antara nilai aktual dan prediksi.
   * MSE lebih sulit diinterpretasikan langsung karena satuannya kuadrat (*dolar kuadrat*). Metrik ini memberikan bobot yang lebih besar pada kesalahan prediksi yang besar (nilai outlier pada residual). Angka yang besar ini mencerminkan adanya beberapa prediksi dengan selisih yang cukup signifikan.
5. **RMSE (Root Mean Squared Error): $5,956.34**
   * RMSE sebesar **$5,956.34** adalah akar kuadrat dari MSE dan memiliki satuan yang sama dengan variabel target (dolar).
   * RMSE sering menjadi metrik evaluasi utama karena satuannya yang sama dengan target dan sensitivitasnya terhadap kesalahan besar. Nilai **$5,956.34** menunjukkan bahwa, rata-rata, model ini membuat kesalahan prediksi sekitar **$5,956.34**, dengan mempertimbangkan bahwa kesalahan yang lebih besar mendapat penalti yang lebih berat (dibandingkan MAE **$4,177.05**).

**Kesimpulan Evaluasi**  
Model Regresi Linier ini menunjukkan performa yang cukup baik dengan **R-squared 0.807**, namun masih memiliki beberapa keterbatasan:

* **Akurasi Prediksi**:
  + MAPE **41.40%** menunjukkan rata-rata kesalahan prediksi yang cukup besar.
  + Perbedaan signifikan antara MAE (**$4,177.05**) dan RMSE (**$5,956.34**) mengindikasikan adanya outlier.
* **Rekomendasi Perbaikan**:
  + Eksplorasi fitur tambahan yang mungkin berpengaruh.
  + Penanganan outlier yang lebih baik.
  + Pertimbangan untuk menggunakan model alternatif yang lebih kompleks.
* **Kelayakan Model**:
  + Meskipun ada kesalahan prediksi, model ini sudah dapat memberikan perkiraan biaya asuransi dengan tingkat keakuratan yang layak untuk kasus penggunaan tertentu.
  + Cocok sebagai *baseline model* sebelum pengembangan lebih lanjut.

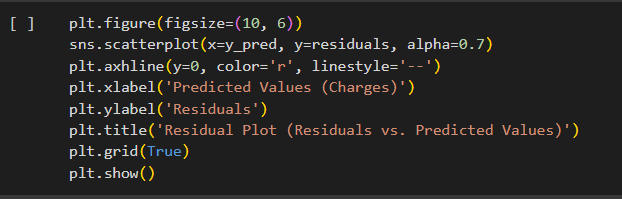
### 2.6.4. Analisis Residual

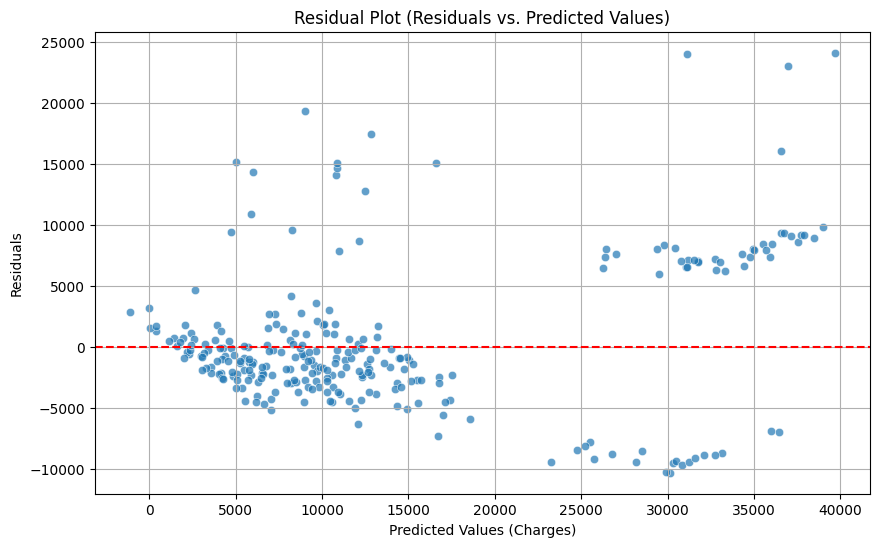
Setelah melatih model Regresi Linier dan mengevaluasi performanya menggunakan metrik seperti R-squared dan MAE/RMSE, langkah penting selanjutnya adalah memeriksa residual. Residual adalah selisih antara nilai biaya asuransi yang sebenarnya (y\_test) dengan nilai biaya asuransi yang diprediksi oleh model (y\_pred) untuk setiap individu dalam data pengujian.

Analisis residual sangat penting karena residual seharusnya tidak menunjukkan pola yang jelas jika model regresi linier sudah sesuai. Jika ada pola dalam residual, ini bisa mengindikasikan bahwa asumsi model regresi linier tidak terpenuhi atau ada informasi penting dalam data yang belum berhasil ditangkap oleh model.

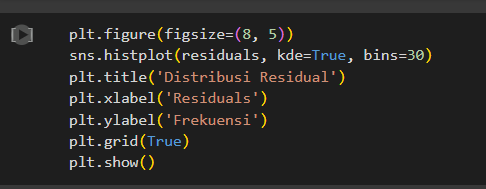


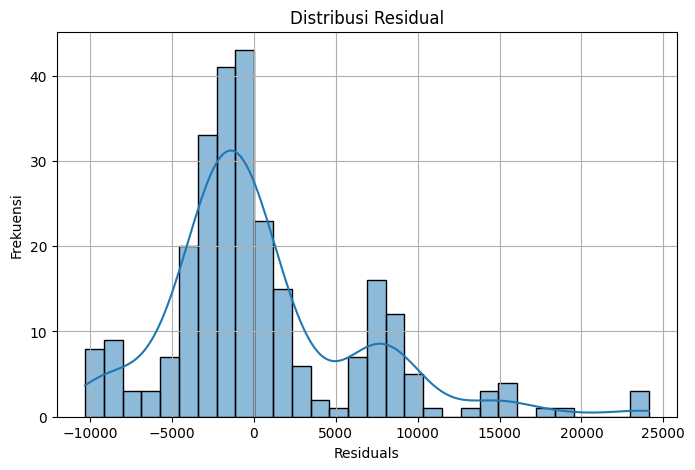
**2.6.4.1. Plot Residual vs. Predicted Values**



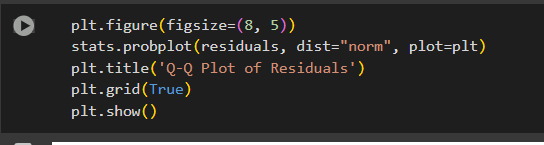


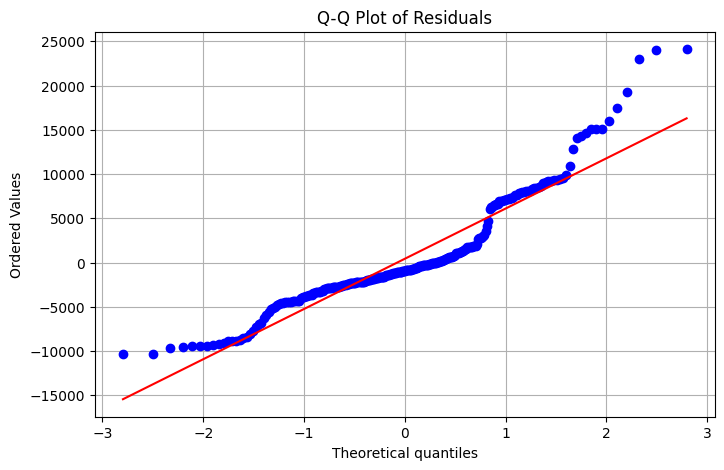
**2.6.4.2. Histogram Residual**



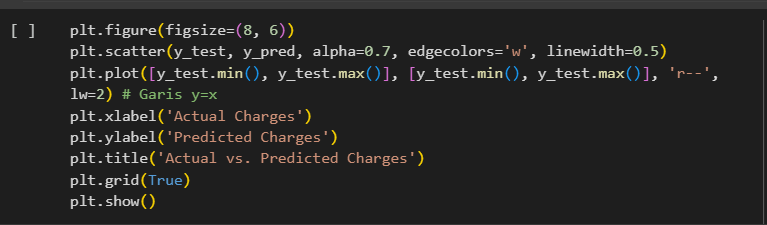


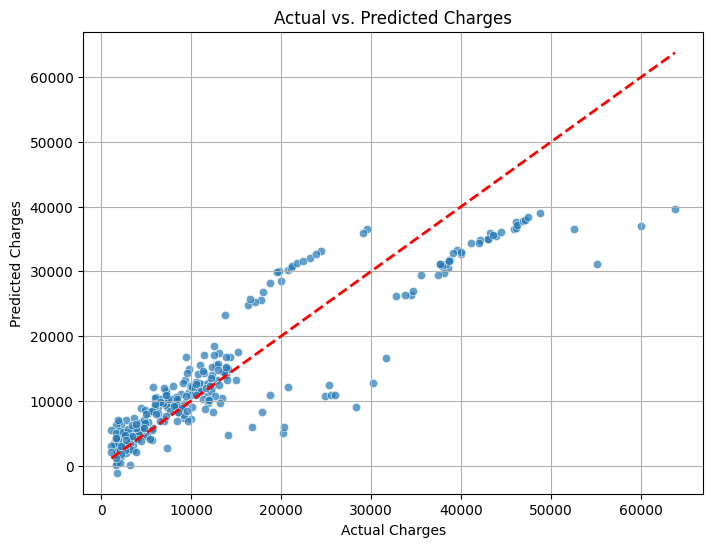
**2.6.4.3. Q-Q Plot Residual**



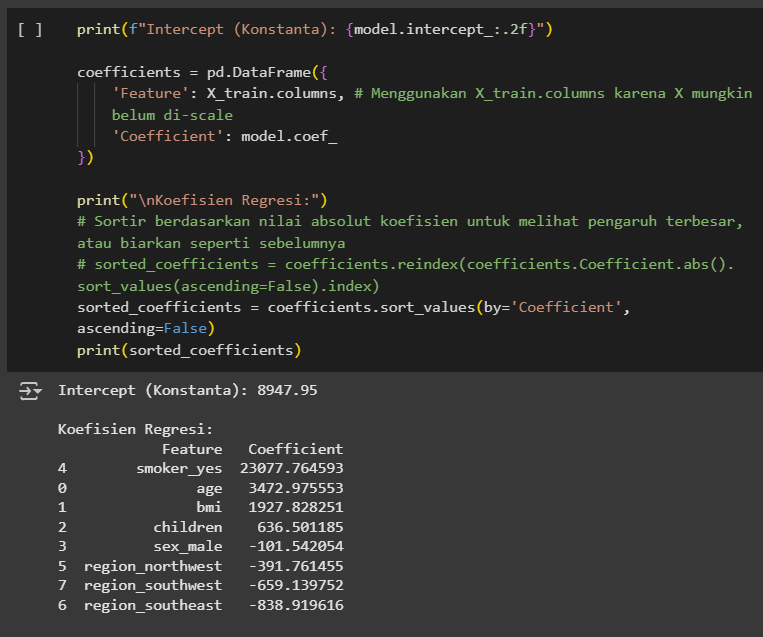


**2.6.4.4. Plot Actual vs. Predicted Values**

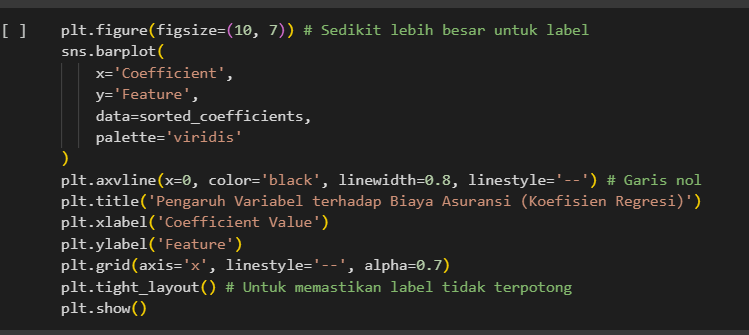


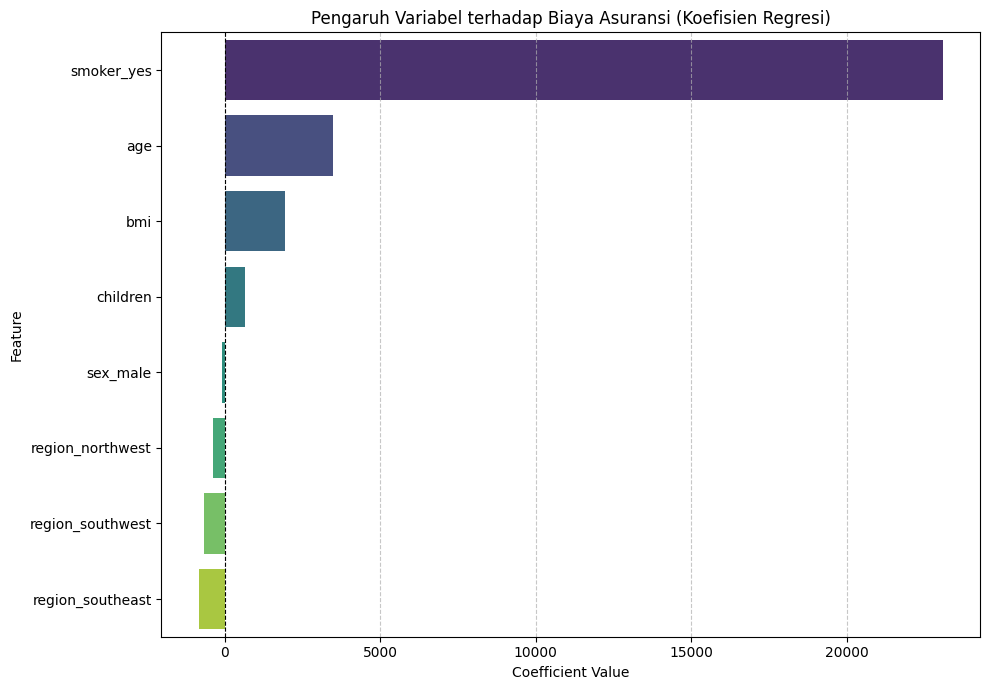


**2.6.7. Tampilkan Koefisien dan Intercept**

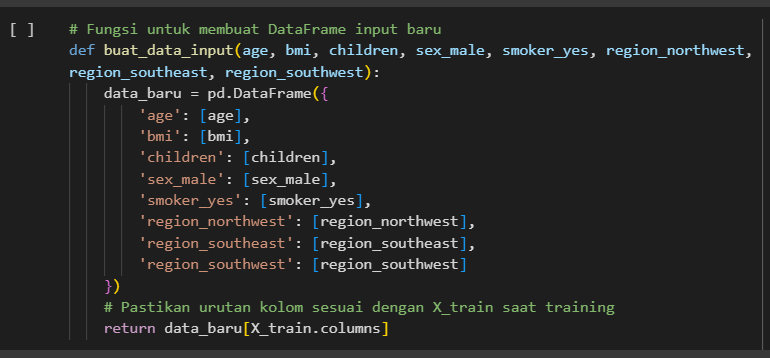


Visualisasi Koefisien yang Ditingkatkan

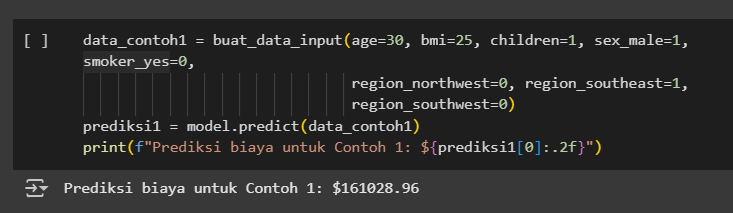




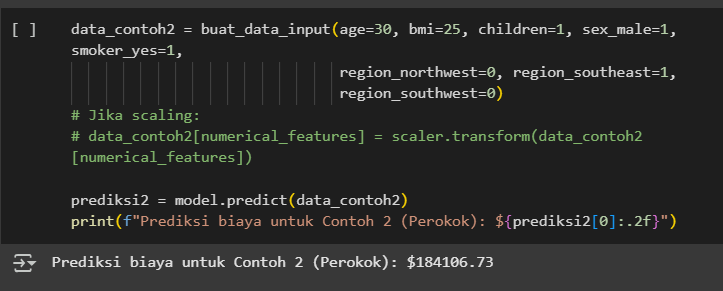
## 2.7. Simulasi Prediksi



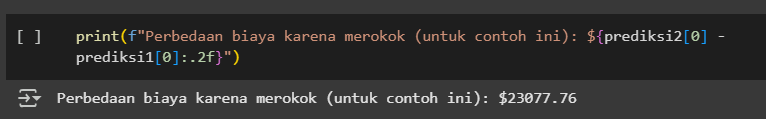
### 2.7.1. Contoh 1: Non-perokok, laki-laki, 30 tahun, bmi 25, 1 anak, di southeast



### 2.7.2. Contoh 2: Sama seperti contoh 1, tetapi perokok



### 2.7.3. Perbedaan biaya karena merokok (untuk contoh ini)



# Bab 3: Kesimpulan

Berdasarkan analisis yang telah dilakukan dalam bab sebelumnya, dapat ditarik beberapa kesimpulan penting terkait model prediksi biaya asuransi kesehatan:

1. **Pengembangan Model dan Kinerja**

Telah berhasil dibangun sebuah model regresi linier yang mampu memprediksi biaya asuransi kesehatan individu. Model ini menunjukkan kinerja yang cukup baik dalam menjelaskan variasi biaya asuransi, sebagaimana tercermin dari nilai R-squared sebesar 0.807. Artinya, sekitar 80.7% variabilitas dalam biaya asuransi dapat dijelaskan oleh variabel-variabel yang digunakan dalam model. Meskipun demikian, nilai MAPE (Mean Absolute Percentage Error) sebesar 41.40% mengindikasikan bahwa rata-rata persentase kesalahan prediksi masih cukup signifikan.

1. **Faktor-Faktor Penting**

Analisis koefisien regresi menunjukkan bahwa status merokok merupakan faktor yang paling signifikan dalam mempengaruhi biaya asuransi, dengan koefisien positif yang besar. Variabel lain seperti usia dan indeks massa tubuh (BMI) juga menunjukkan pengaruh positif terhadap peningkatan biaya asuransi. Sebaliknya, beberapa kategori wilayah tempat tinggal menunjukkan korelasi negatif dengan biaya, mengindikasikan potensi biaya yang lebih rendah dibandingkan dengan wilayah referensi.

1. **Proses Data**

Keberhasilan model ini didukung oleh serangkaian tahapan pra-pemrosesan data yang cermat. Ini termasuk penanganan data duplikat, pengkodean (encoding) variabel-variabel kategorikal seperti jenis kelamin, status perokok, dan wilayah menjadi format numerik. Selain itu, dilakukan penskalaan (scaling) pada fitur-fitur numerik untuk memastikan semua variabel memberikan kontribusi yang setara dalam model.

1. **Validasi Model**

Evaluasi model dilakukan menggunakan metrik standar seperti MAE (Mean Absolute Error) sebesar $4,177.05 dan RMSE (Root Mean Squared Error) sebesar $5,956.34. Perbedaan antara MAE dan RMSE menyiratkan adanya beberapa prediksi dengan kesalahan yang cukup besar, yang juga terlihat dari analisis residual. Plot residual menunjukkan adanya beberapa pola, seperti heteroskedastisitas, yang mengindikasikan bahwa varians kesalahan tidak konstan di semua tingkat nilai prediksi.

1. **Implikasi dan Rekomendasi**

Model yang dikembangkan dapat memberikan estimasi awal biaya asuransi dan menjadi dasar untuk analisis lebih lanjut. Meskipun memiliki kemampuan prediktif yang baik, terdapat ruang untuk perbaikan. Upaya di masa mendatang dapat difokuskan pada eksplorasi fitur tambahan, penanganan outlier yang lebih cermat, atau pertimbangan untuk menggunakan model machine learning alternatif yang mungkin lebih kompleks guna meningkatkan akurasi prediksi dan mengatasi pola yang teramati dalam residual.

Secara keseluruhan, proyek ini berhasil menunjukkan penerapan regresi linier dalam memprediksi biaya asuransi dan mengidentifikasi faktor-faktor kunci yang mempengaruhinya, seraya memberikan landasan untuk pengembangan model yang lebih lanjut.

# Bab 4: Lampiran

Pembagian tugas:

* Lathif Ramadhan (5231811022) : Pembuatan kode dan penyusunan serta merapikan laporan
* Andini Angel M. (5231811029) : Pembuatan kode
* Rama Panji N. (5231811033) : Penyusunan laporan
* Giffari Riyanda P. (5231811036) : Pembuatan laporan