NAME: ABEL

SURENAME: ILUNGA

STUDENT NUMBER: ST10090262 POE PART 1

MODULE: PDAN8411

LINEAR REGRESSION REPORT: MEDICAL CHARGES A PREDICTIONS

INTRODUCTION

The goal of medical assistance programs is to provide customers with reasonable, data-driven rates. This project offers a data-driven proof of concept that predicts medical insurance costs based on lifestyle and demographic parameters using linear regression. By identifying the main factors influencing medical expenses, the model will assist the business in creating more precise pricing plans that are suited to the needs of individual clients.

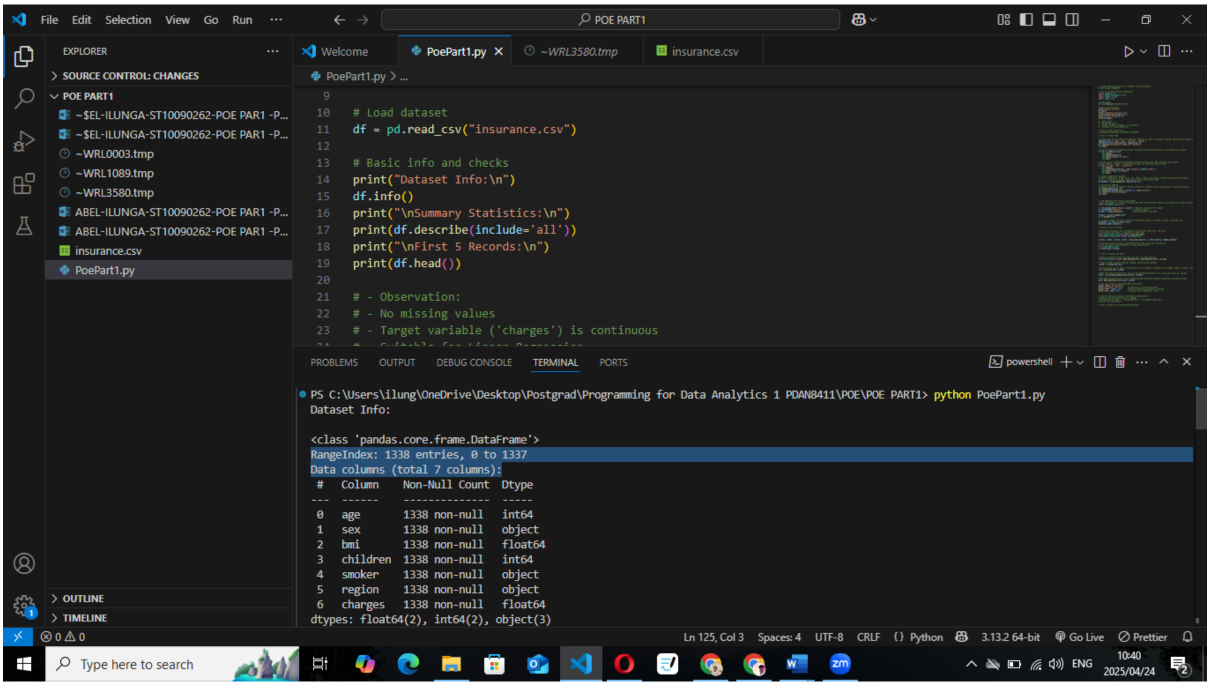
1. DATA OVERVIEW

The datasets consist of **1338** records with **7** features including demographics and lifestyles variables such

- age: Age of the individual
- sex: Gender (binary)
- bmi: Body Mass Index
- children: Number of children covered
- smoker: Smoking status
- region: Geographic region in the US
- charges: Annual medical charges (target variable)

1.1 There are no mussing values
1.2 The target variables (charges) is continuous, ideal for linear regression.
1.3 Categorical feature like sex, smoker, and region will be encouraged.
1.4 Suitability confirmed based on the variable types and structure.

Code: to be found in the Jupyter file



```python

# Load dataset
df = pd.read_csv("insurance.csv")

# Basic info and checks
print("Dataset Info:\n")
df.info()
print("\nSummary Statistics:\n")
print(df.describe(include='all'))
print("\nFirst 5 Records:\n")
print(df.head())

# - Observation:
# - No missing values
# - Target variable ('charges') is continuous
```

```
PS C:\Users\ilung\OneDrive\Desktop\Postgrad\Programming for Data Analytics 1 PDAN8411\POE\POE PART1> python PoePart1.py
Dataset Info:

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1338 entries, 0 to 1337
Data columns (total 7 columns):
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   age       1338 non-null   int64
 1   sex       1338 non-null   object
 2   bmi       1338 non-null   float64
 3   children  1338 non-null   int64
 4   smoker    1338 non-null   object
 5   region    1338 non-null   object
 6   charges   1338 non-null   float64
dtypes: float64(2), int64(2), object(3)
```

2. PLAN OF ANALYSIS

I created a precise analysis plan of creating a strong linear regression model to forecast medical insurance costs, making sure that every stage is rationally organised and in line with the modelling goals. This plan is visually summarised in the flowchart below, which is divided into five main stages:

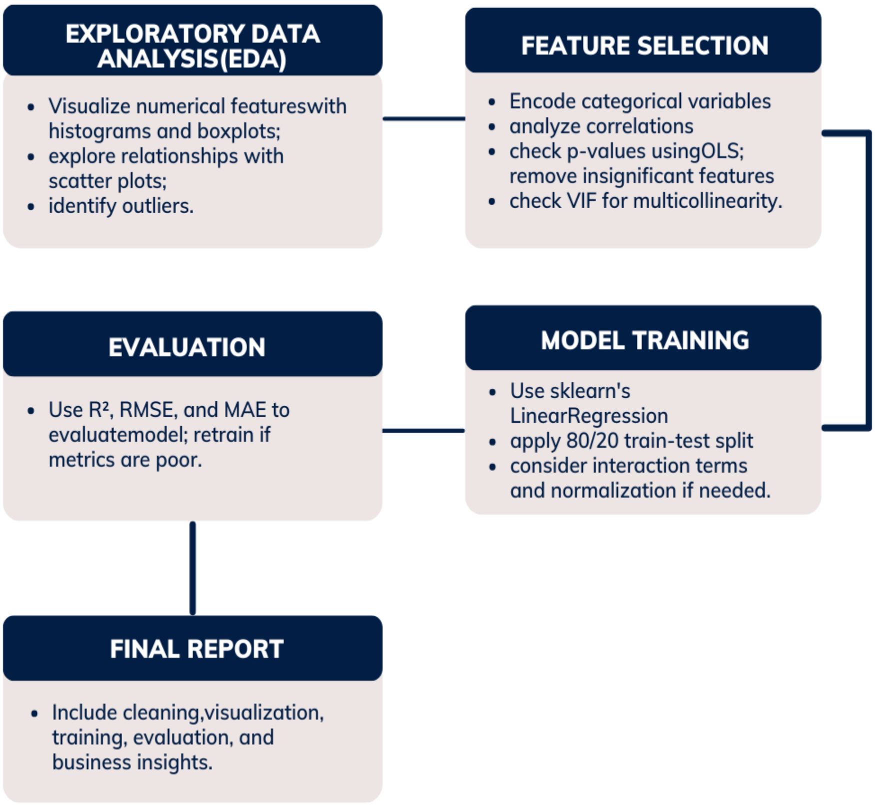
a. **Exploratory Data Analysis (EDA)**: This phase includes visualizing distributions of numeric variables (e.g., age, BMI, charges), identifying outliers, and exploring relationships between features (e.g., smoking vs. charges). This step ensures data quality and highlights patterns.

b. **Feature Selection**: After encoding categorical features (e.g., sex, region), I will assess correlation, use OLS p-values to test significance, and check multicollinearity with VIF. Irrelevant or redundant features will be removed to improve model efficiency and interpretability.

c. **Model Training**: The model will be trained using a train-test split (typically 80/20). Scikit-lean's Linear Regression will be used initially, with potential tuning or transformation (e.g., normalization or interaction terms) if needed.

d. **Model Evaluation**: Key evaluation metrics will include R-squared (for variance explanation), RMSE (for error magnitude), and MAE (for average error). These will guide any retraining or adjustments.

e. **Report and Insights**: Finally, the results will be compiled into a business-focused report that includes EDA findings, model interpretation, and insights relevant to the medical aid client's goals.



EXPLORATORY DATA ANALYSIS(EDA)

- Visualize numerical featureswith histograms and boxplots;
- explore relationships with scatter plots;
- identify outliers.

FEATURE SELECTION

- Encode categorical variables
- analyze correlations
- check p-values usingOLS; remove insignificant features
- check VIF for multicollinearity.

MODEL TRAINING

- Use sklearn's LinearRegression
- apply 80/20 train-test split
- consider interaction terms and normalization if needed.

EVALUATION

- Use R^2, RMSE, and MAE to evaluatemodel; retrain if metrics are poor.

FINAL REPORT

- Include cleaning,visualization, training, evaluation, and business insights.

3. CONDUCTING THE ANALYSIS

In this section, I use Python to illustrate the entire analysis pipeline. To guarantee openness, reproducibility, and conformity to regression modelling best practices, each step is explicably explained and supported.

Exploratory Data Analysis (EDA)

- Histograms are useful for visualising continuous variable distributions and identifying uniformity or skewness. (Kenton, 2022).
- To identify outliers and comprehend how data is distributed across quartiles, boxplots are utilised. (McKinney, 2018).
- The hue='smoker' property in scatter plots draws attention to possible correlations between factors and insurance premiums:
    - Smokers generally incur higher charges.
    - Age correlates positively with medical charges.
- A correlation heatmap aids in the detection of multicollinearity by visualising the direction and intensity of linear correlations between variables. (Seaborn Docs, 2023).

Feature Selection

- For model compatibility, one-hot encoding is used to transform categorical variables (sex, region, and smoker) into binary features. (Scikit-learn, 2023).
- OLS Regression is used to evaluate the statistical significance of features through p-values, identifying which features significantly impact the target (charges) (Brooks et al., 2019).

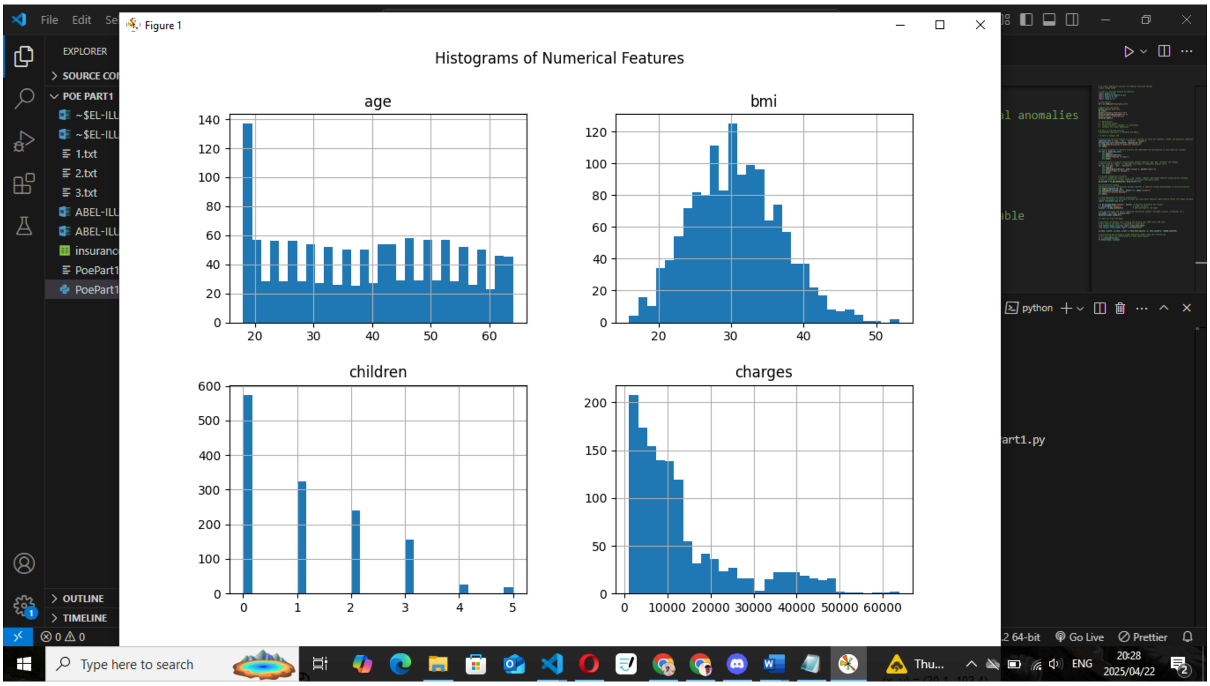
Model Training

- By conserving unseen data for testing, an 80/20 train-test split guarantees equitable model evaluation. (Scikit-learn, 2023).
- Scikit-learn is used to construct linear regression, which minimises the residual sum of squares between observed and expected values to estimate coefficients. (James et al., 2013).

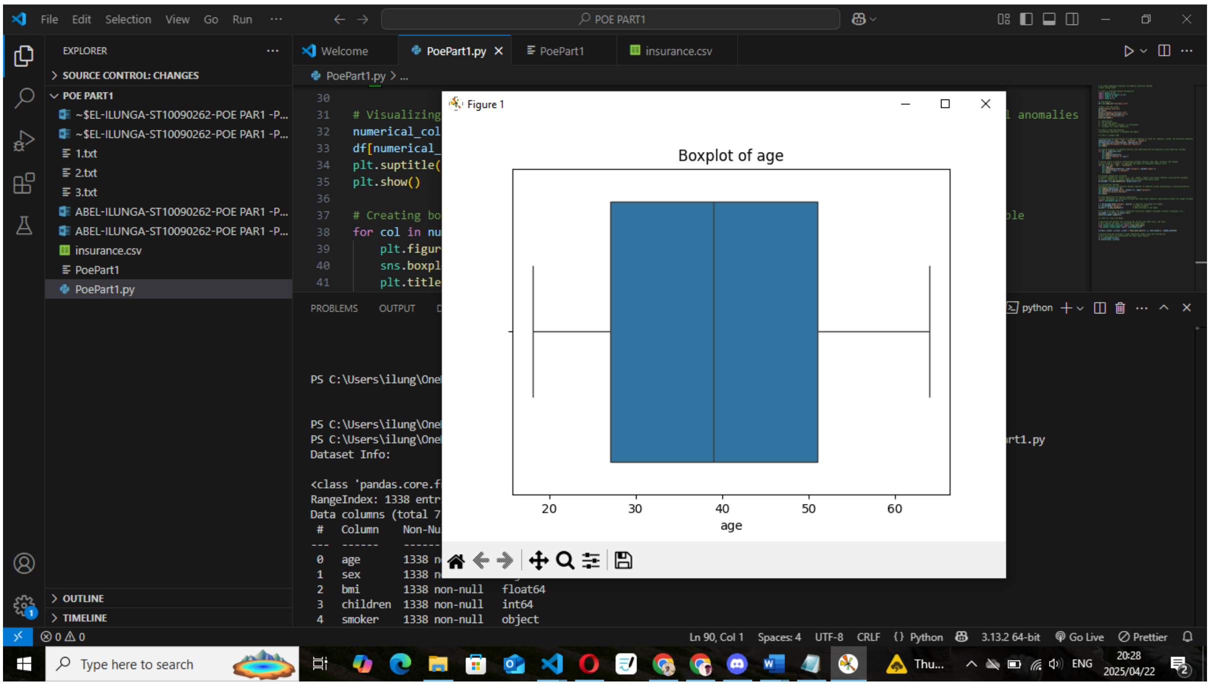
This model acts as a baseline, even though basic linear regression lacks hyperparameters that can be adjusted. If assessment measures indicate underperformance, normalisation or interaction terms can be investigated later.

Each of these steps contributes to building a transparent, accurate, and explainable model tailored for predicting medical insurance charges, the following images demonstrates all details explained above, and more steps on the achievements are in the code *Jupyter* submitted along with this report

Histograms



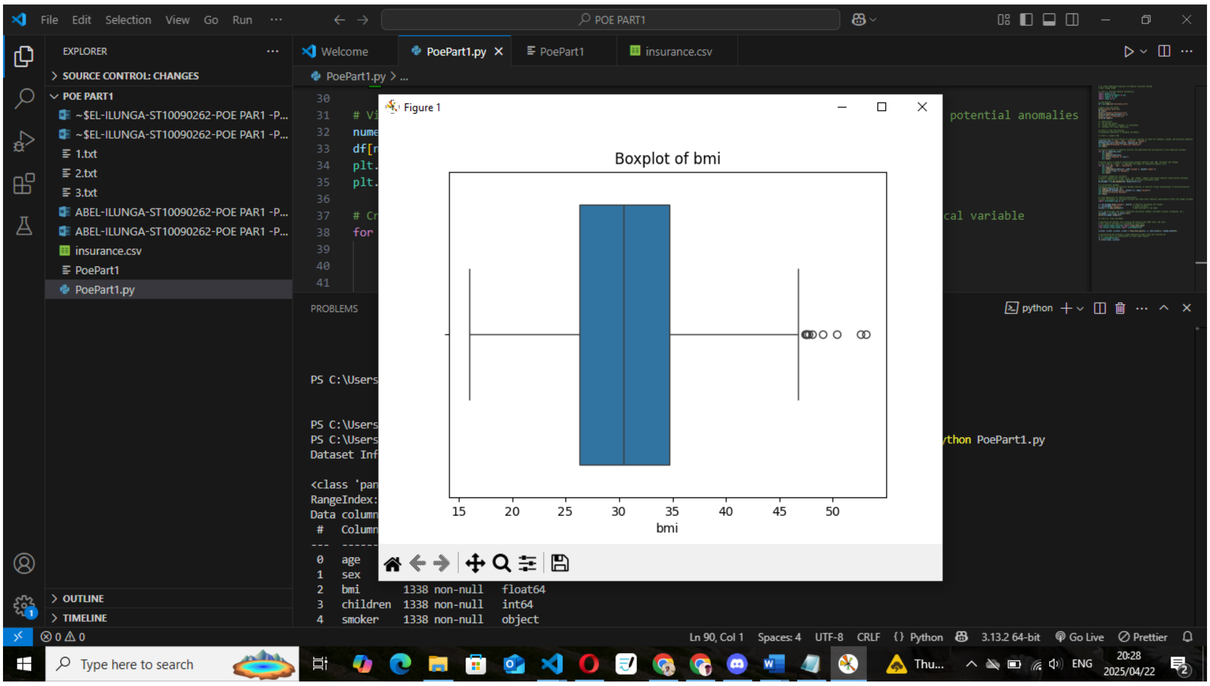
Boxplots (age)



```
# Visualizing                    l anomalies
numerical_col
df[numerical_
plt.suptitle(
plt.show()

# Creating bo                    ble
for col in nu
    plt.figur
    sns.boxpl
    plt.title
```

```
PS C:\Users\ilung\One


PS C:\Users\ilung\One                    rt1.py
PS C:\Users\ilung\One
Dataset Info:

<class 'pandas.core.f
RangeIndex: 1338 entr
Data columns (total 7
 #   Column    Non-Nu
---  ------    ------
 0   age       1338 n
 1   sex       1338 n
 2   bmi       1338 non-null   float64
 3   children  1338 non-null   int64
 4   smoker    1338 non-null   object
```

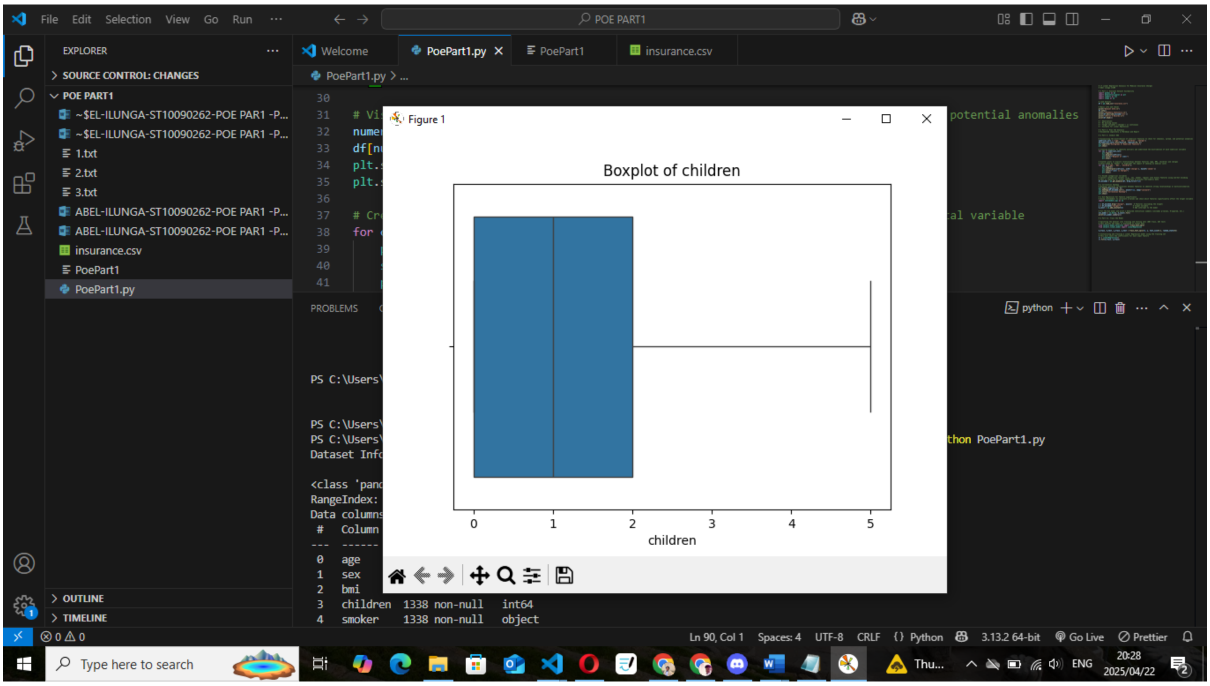
Boxplots (bmi)



```
# Vi                    potential anomalies
nume
df[r
plt.
plt.

# Cr                    cal variable
for
```

```
PS C:\Users


PS C:\Users                    thon PoePart1.py
PS C:\Users
Dataset Inf

<class 'pan
RangeIndex:
Data column
 #   Column
---  ------
 0   age
 1   sex
 2   bmi       1338 non-null   float64
 3   children  1338 non-null   int64
 4   smoker    1338 non-null   object
```

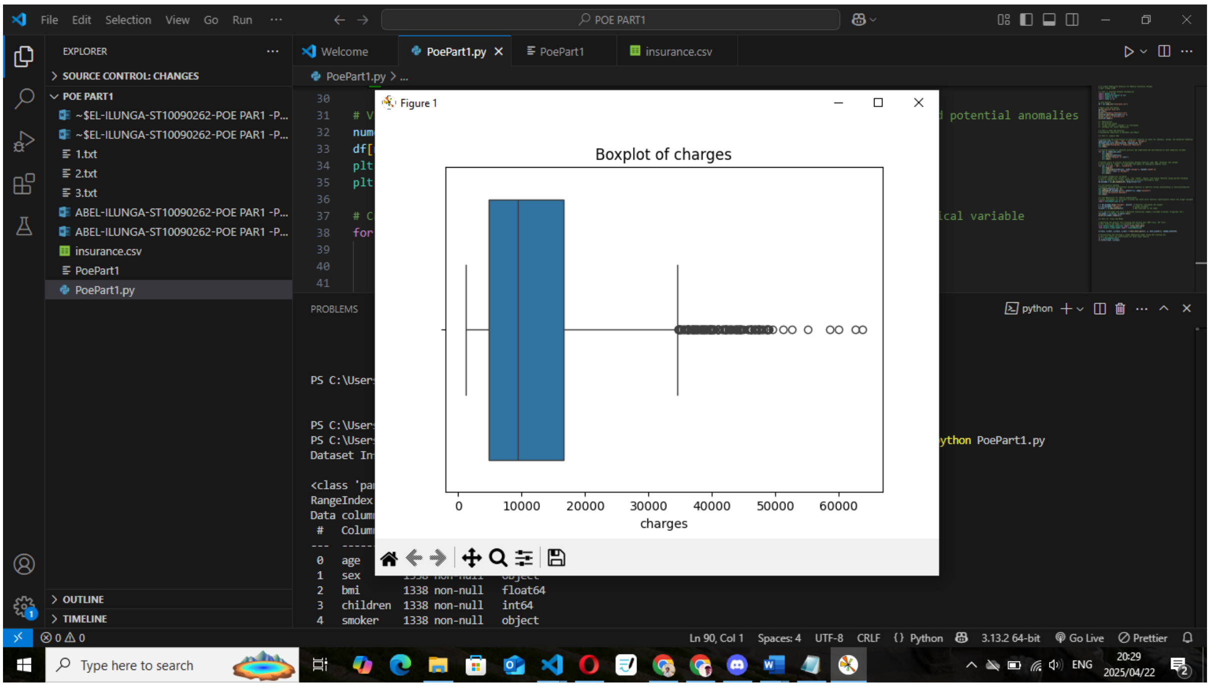
Boxplots (children)



```
# Vi                                   potential anomalies
nume
df[nu
plt.
plt.

# Cr                                   al variable
for
```

```
PS C:\Users\


PS C:\Users\
PS C:\Users\                           thon PoePart1.py
Dataset Info

<class 'pand
RangeIndex:
Data columns
 #   Column
---  ------
 0   age
 1   sex
 2   bmi
 3   children  1338 non-null   int64
 4   smoker    1338 non-null   object
```

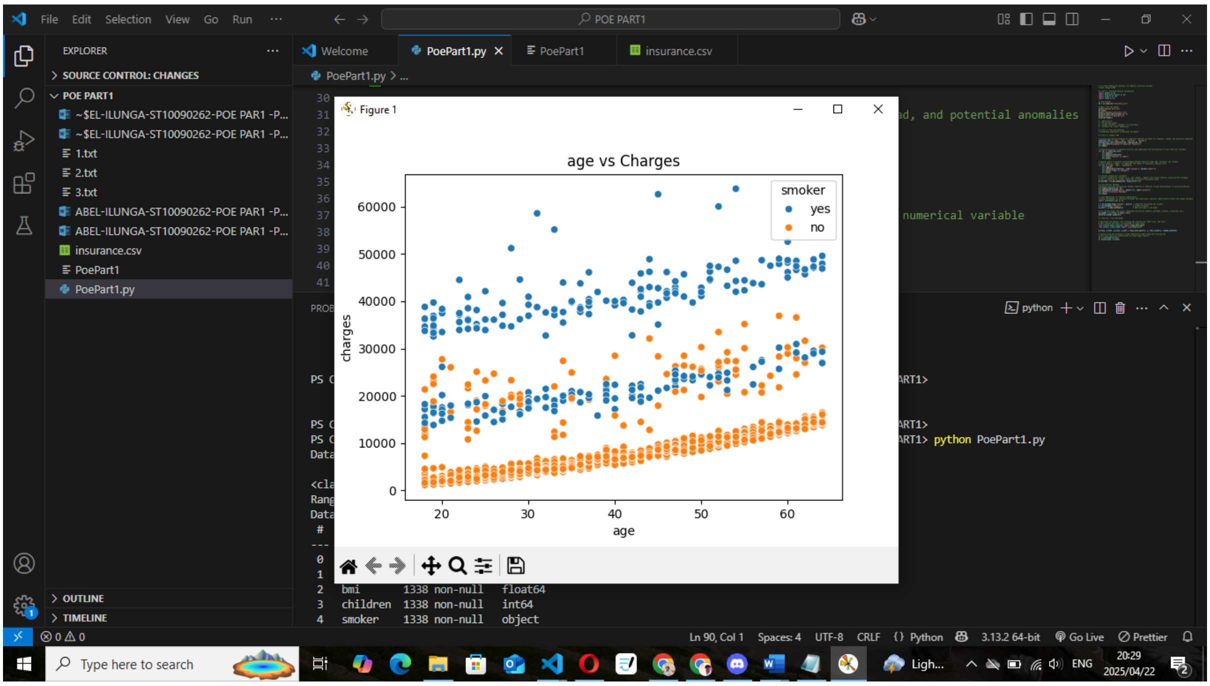
Boxplots(charges)



```
# V                                    d potential anomalies
num
df[
plt
plt

# C                                    ical variable
for
```

```
PS C:\User


PS C:\User
PS C:\User                             ython PoePart1.py
Dataset In

<class 'pa
RangeIndex
Data colum
 #   Colum
---  -----
 0   age
 1   sex
 2   bmi       1338 non-null   float64
 3   children  1338 non-null   int64
 4   smoker    1338 non-null   object
```

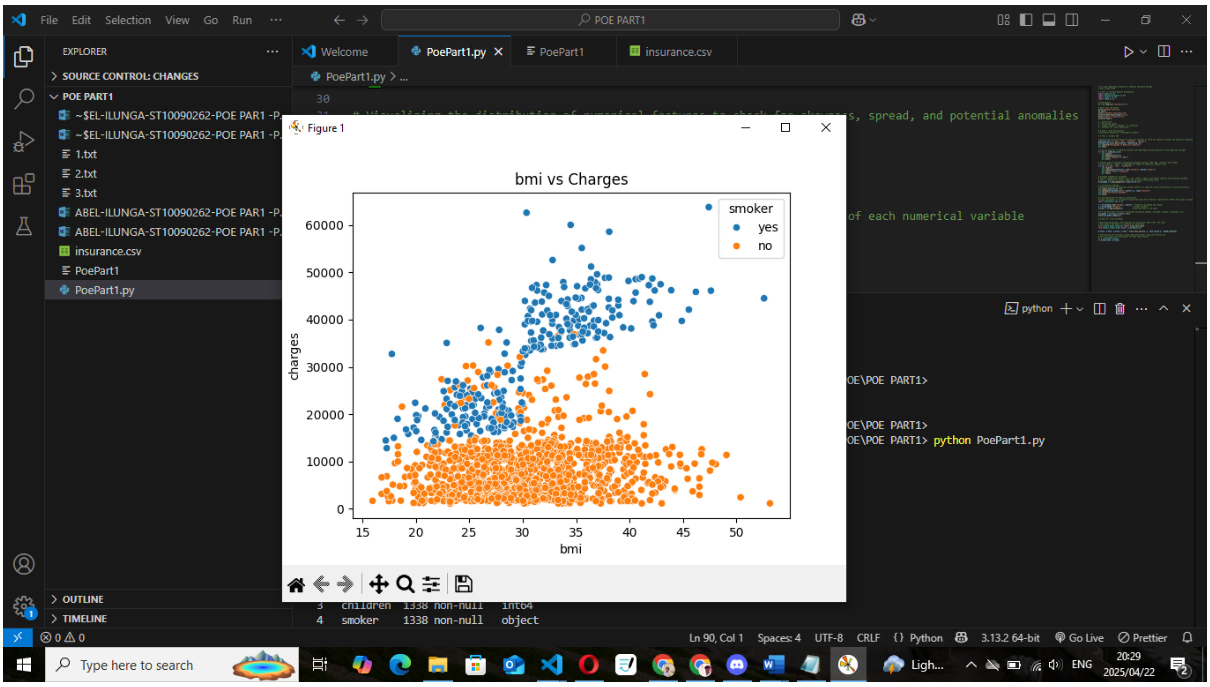
Scatter Plots (age vs. charges)



```
ad, and potential anomalies
numerical variable
```

```
PS C
                                                     ART1>

PS C                                                 ART1>
PS C                                                 ART1> python PoePart1.py
Data

<cla
Rang
Data
 #
---
 0
 1
 2   bmi       1338 non-null   float64
 3   children  1338 non-null   int64
 4   smoker    1338 non-null   object
```

Scatter Plots(bmi vs charges)

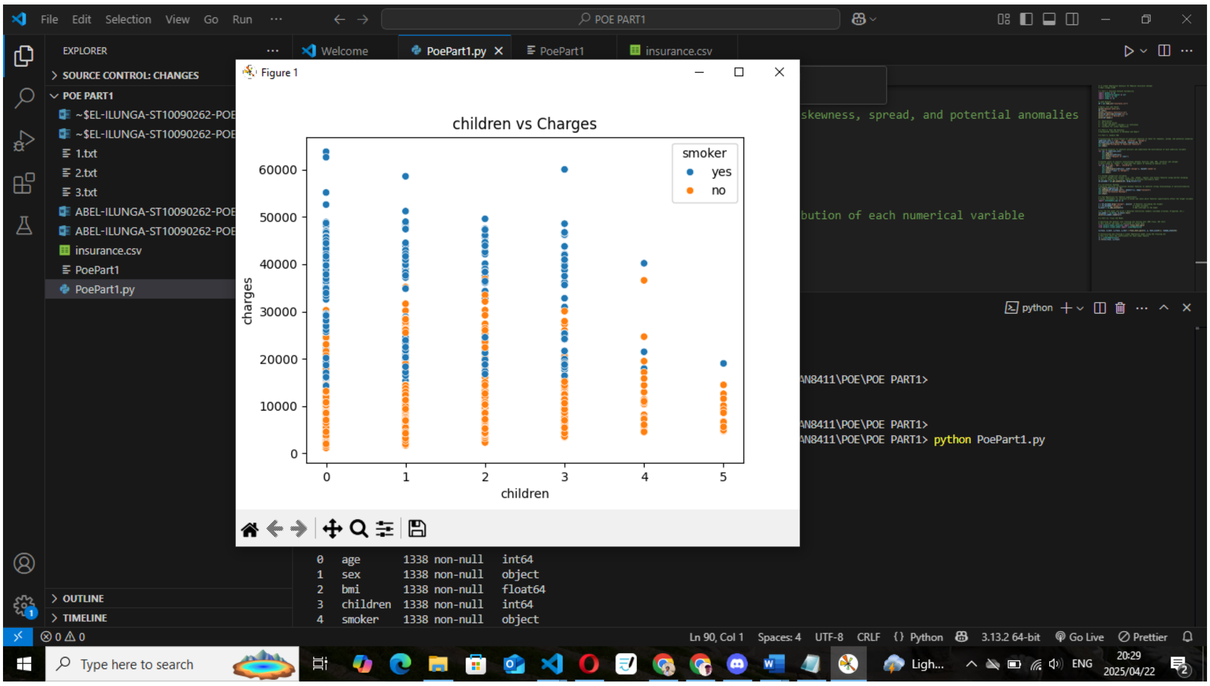


```
# Visualizing the distribution of numerical features to check for skewness, spread, and potential anomalies
of each numerical variable
```

```
OE\POE PART1>

OE\POE PART1>
OE\POE PART1> python PoePart1.py
```

```
 3   children  1338 non-null   int64
 4   smoker    1338 non-null   object
```

Scatter Plots (children vs charges)

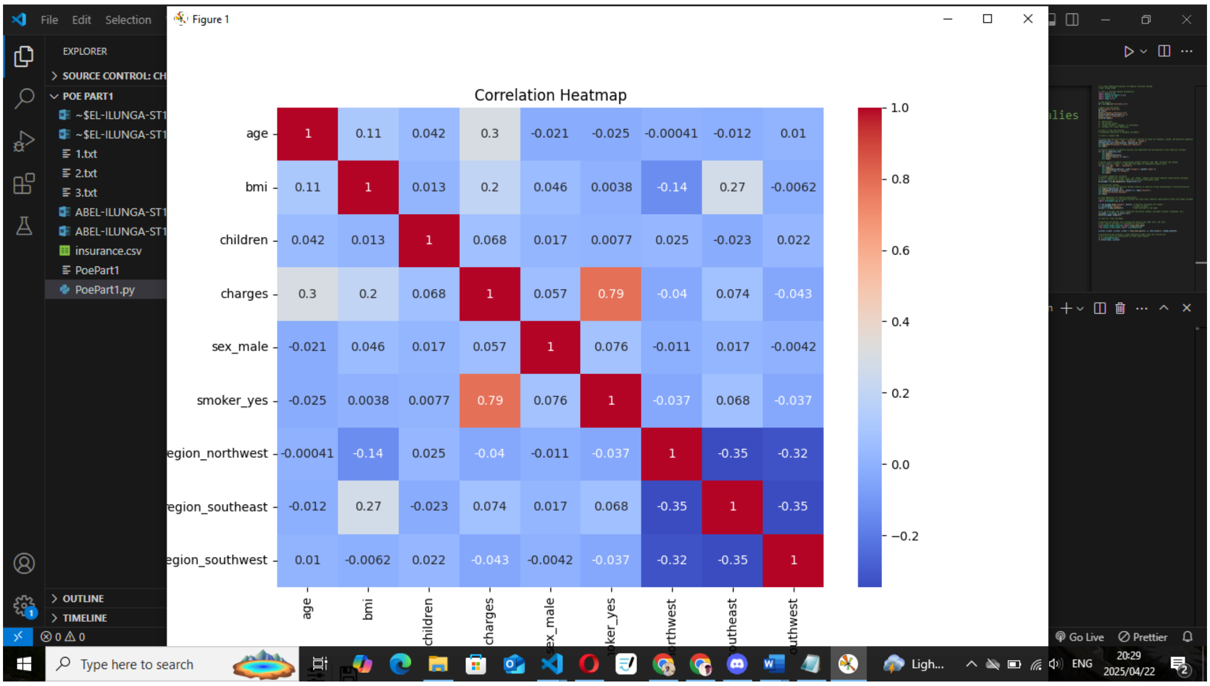


```
skewness, spread, and potential anomalies

bution of each numerical variable
```

```
AN8411\POE\POE PART1>

AN8411\POE\POE PART1>
AN8411\POE\POE PART1> python PoePart1.py
```

```
 0   age       1338 non-null   int64
 1   sex       1338 non-null   object
 2   bmi       1338 non-null   float64
 3   children  1338 non-null   int64
 4   smoker    1338 non-null   object
```

Correlation Heatmap



	age	bmi	children	charges	ex_male	ker_yes	rthwest	utheast	uthwest
age	1	0.11	0.042	0.3	-0.021	-0.025	-0.00041	-0.012	0.01
bmi	0.11	1	0.013	0.2	0.046	0.0038	-0.14	0.27	-0.0062
children	0.042	0.013	1	0.068	0.017	0.0077	0.025	-0.023	0.022
charges	0.3	0.2	0.068	1	0.057	0.79	-0.04	0.074	-0.043
sex_male	-0.021	0.046	0.017	0.057	1	0.076	-0.011	0.017	-0.0042
smoker_yes	-0.025	0.0038	0.0077	0.79	0.076	1	-0.037	0.068	-0.037
egion_northwest	-0.00041	-0.14	0.025	-0.04	-0.011	-0.037	1	-0.35	-0.32
egion_southeast	-0.012	0.27	-0.023	0.074	0.017	0.068	-0.35	1	-0.35
egion_southwest	0.01	-0.0062	0.022	-0.043	-0.0042	-0.037	-0.32	-0.35	1

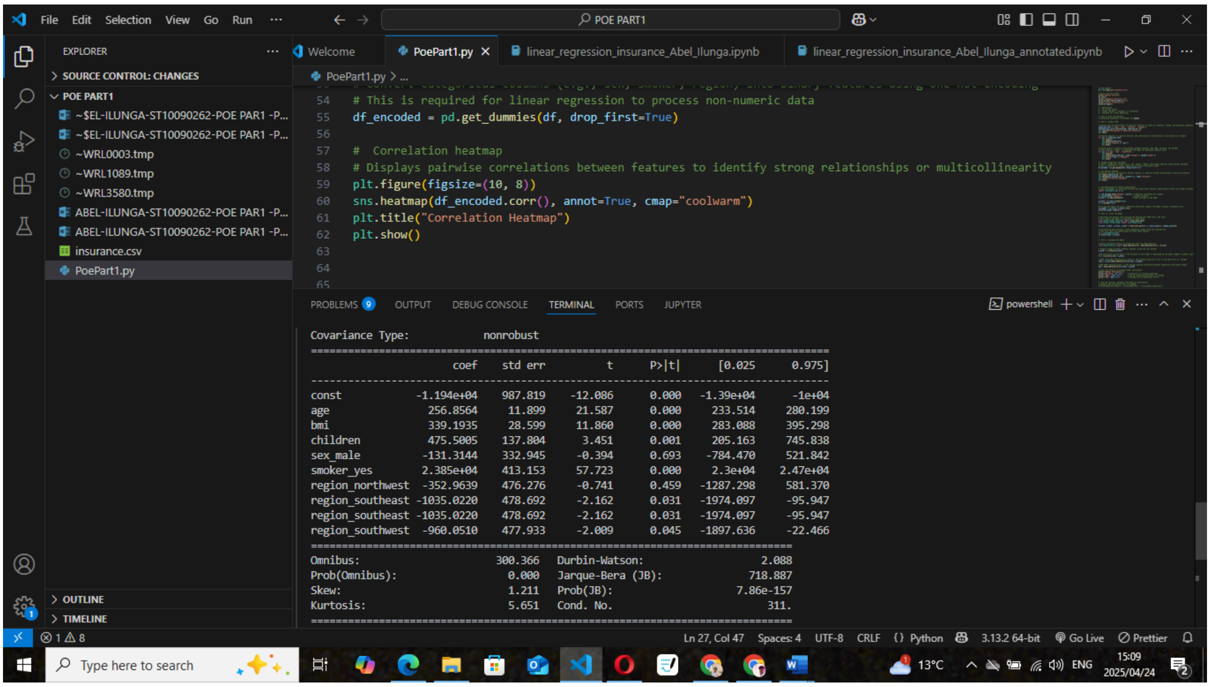
4. EVALUATING THE MULTIPLE LINEAR REGRESSION MODEL

Model Summary

I used Ordinary Least Squares (OLS) to train a Multiple Linear Regression model after preprocessing the data, which included one-hot encoding categorical variables and dividing the dataset into training and testing sets. The model overview is as follows:

- Dependent Variable: charges
- R-squared: 0.751
- Adjusted R-squared: 0.749
- F-statistic: 500.8 (p < 0.001)
- Number of Observations: 1338
- Number of Predictors (Df Model): 8

Model Coefficients:



```python
# This is required for linear regression to process non-numeric data
df_encoded = pd.get_dummies(df, drop_first=True)

#  Correlation heatmap
# Displays pairwise correlations between features to identify strong relationships or multicollinearity
plt.figure(figsize=(10, 8))
sns.heatmap(df_encoded.corr(), annot=True, cmap="coolwarm")
plt.title("Correlation Heatmap")
plt.show()
```

```
Covariance Type:            nonrobust
=====================================================================================
                        coef    std err          t      P>|t|      [0.025      0.975]
-------------------------------------------------------------------------------------
const             -1.194e+04    987.819    -12.086      0.000   -1.39e+04      -1e+04
age                 256.8564     11.899     21.587      0.000     233.514     280.199
bmi                 339.1935     28.599     11.860      0.000     283.088     395.298
children            475.5005    137.804      3.451      0.001     205.163     745.838
sex_male           -131.3144    332.945     -0.394      0.693    -784.470     521.842
smoker_yes         2.385e+04    413.153     57.723      0.000     2.3e+04    2.47e+04
region_northwest   -352.9639    476.276     -0.741      0.459   -1287.298     581.370
region_southeast -1035.0220    478.692     -2.162      0.031   -1974.097     -95.947
region_southeast -1035.0220    478.692     -2.162      0.031   -1974.097     -95.947
region_southwest   -960.0510    477.933     -2.009      0.045   -1897.636     -22.466
==============================================================================
Omnibus:                      300.366   Durbin-Watson:                   2.088
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              718.887
Skew:                           1.211   Prob(JB):                    7.86e-157
Kurtosis:                       5.651   Cond. No.                         311.
==============================================================================
```

Interpretation:

According to the regression results, the factors that have the biggest favourable effects on medical charges are age, BMI, and smoking status. For instance, when all other factors are held constant, smoking raises predicted expenses by about R23,850. About

75.1% of the variance in the target variable can be explained by the model, which is regarded as strong for data from the actual world.

Model Evaluation Metrics

To assess the model's predictive performance, I used the following metrics on the test dataset:

- R² Score: 0.784
  This means that 78.4% of the variation in medical charges is explained by the model. This suggests the model has good explanatory power.

- Root Mean Squared Error (RMSE): R5,796.28
  This metric shows the standard deviation of prediction errors. A lower RMSE indicates better model accuracy. Given the wide range of medical expenses, this is a reasonable error level.

- Mean Absolute Error (MAE): R4,181.19
  MAE reflects the average absolute difference between the predicted and actual values. It is easier to interpret and not as sensitive to outliers as RMSE.

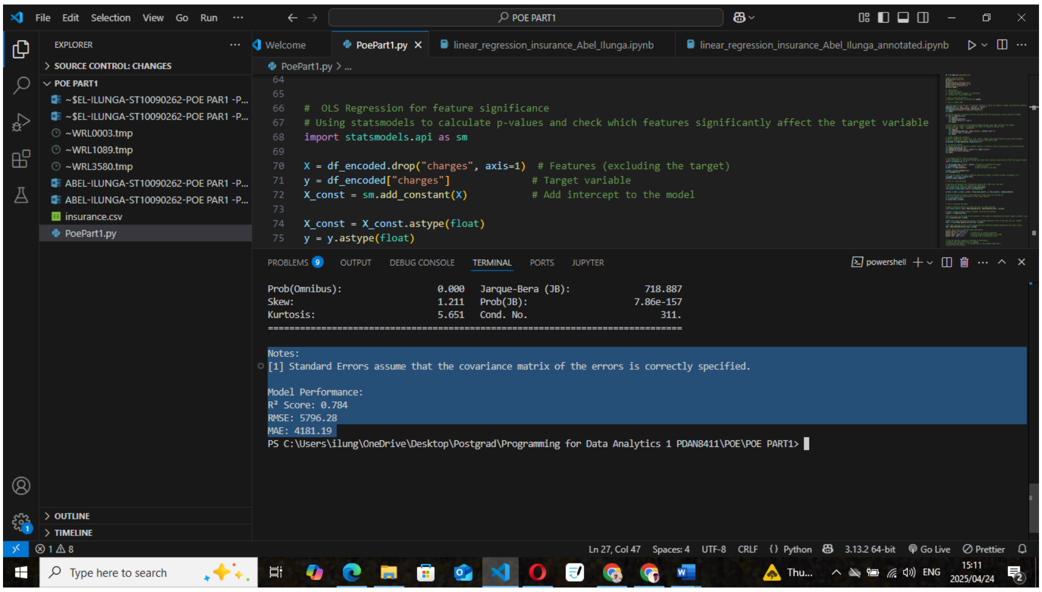
Why these metrics were chosen:

- R² evaluates the model's overall fit.

- RMSE emphasizes larger prediction errors and is good for assessing variance.

- MAE is robust and provides a straightforward measure of average error.

According to *James et al. (2013)* and *Scikit-learn (2023)*, combining these metrics provides a comprehensive view of model performance. Given the results, the model performs well and can be considered reliable for predicting medical charges.



```python


#  OLS Regression for feature significance
# Using statsmodels to calculate p-values and check which features significantly affect the target variable
import statsmodels.api as sm

X = df_encoded.drop("charges", axis=1)  # Features (excluding the target)
y = df_encoded["charges"]              # Target variable
X_const = sm.add_constant(X)           # Add intercept to the model

X_const = X_const.astype(float)
y = y.astype(float)
```

```
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              718.887
Skew:                           1.211   Prob(JB):                    7.86e-157
Kurtosis:                       5.651   Cond. No.                         311.
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

Model Performance:
R² Score: 0.784
RMSE: 5796.28
MAE: 4181.19
PS C:\Users\ilung\OneDrive\Desktop\Postgrad\Programming for Data Analytics 1 PDAN8411\POE\POE PART1>
```

Conclusion on Retraining:
Retraining the model is not necessary currently, according on the existing performance metrics (R2 = 0.784, RMSE = R5,796.28, MAE = R4,181.19). For predicting actual medical costs, the model already offers a high explanatory power and manageable error rates. Future enhancements like regularisation or feature engineering, however, might increase its performance even further.

Reference

Brooks, C., Burke, L., & Persand, G. (2019) The econometrics of financial markets. 3rd edn. Cambridge: Cambridge University Press.

James, G., Witten, D., Hastie, T. and Tibshirani, R. (2013) An introduction to statistical learning: with applications in R. New York: Springer.

Kenton, W. (2022) Histogram Definition. Investopedia. Available at: https://www.investopedia.com/terms/h/histogram.asp (Accessed: 24 April 2025).

McKinney, W. (2018) Python for data analysis: data wrangling with Pandas, NumPy, and IPython. 2nd edn. Sebastopol, CA: O'Reilly Media.

Scikit-learn (2023) Scikit-learn: Machine Learning in Python. Available at: https://scikit-learn.org/stable/ (Accessed: 24 April 2025).

Seaborn Docs (2023) Seaborn: Statistical data visualization. Available at: https://seaborn.pydata.org/ (Accessed: 24 April 2025).

Choi, M. (2018) *Insurance Dataset*. Kaggle. Available at: https://www.kaggle.com/datasets/mirichoi0218/insurance (Accessed: 24 April 2025).