4/25/2025

Programming for Data Analytics 1 - Formative part 1

PDAN8411

ST10036066 Zanokuhle Azania Ncube

VARSITY COLLEGE CAPE TOWN

Table of Contents

Table of Figures

Introduction

A South African medical aid scheme requires tailored medical aid costs according to the lifestyle choices and client attributes such as age and body mass index measurements. In data analysis, the is a process followed to gain business insight and to solve a problem. The steps are known as the Data Analysis Process as it aids as a template to correctly find a problem followed by the data collection, cleaning, preparations, analysis and visualisations to help an organisation such as the South African medical aid health insurance. This document consists of how the data analysis process has been practically implemented to help the South African medical aid scheme. Each sub-heading is a step within the data analysis process and the practical implementations. Finally, the report concludes which Linear Regression model predicted the best score.

Identifying and understanding the South African Medical aid scheme problem

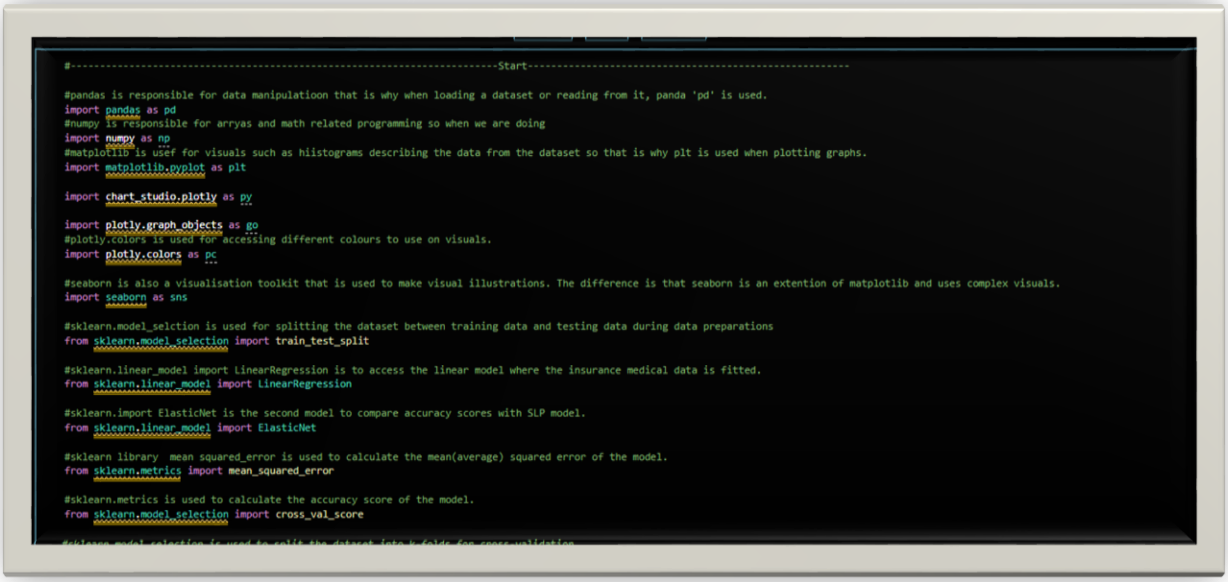
The requirement was specified by the organisation that a linear Regression model needs to be used to adjust and tailor their charges for their clients according to their lifestyle choice therefore a Kaggle dataset has been provided for the data analysis.

The Data Collection process

The dataset is publicly available on Kaggle. The data used is based from United States of American health insurance clients' attributes and lifestyle choices. It consist of 1338 data points. The dataset represents a medical aid scheme client data such as the age of the client who is the primary beneficiary, their gender, Body Mass Index, geographic residential regional location, the number of children/dependents, smoking status and the costs of billed health insurance charges. (Choi, 2018)

Using Python libraries.

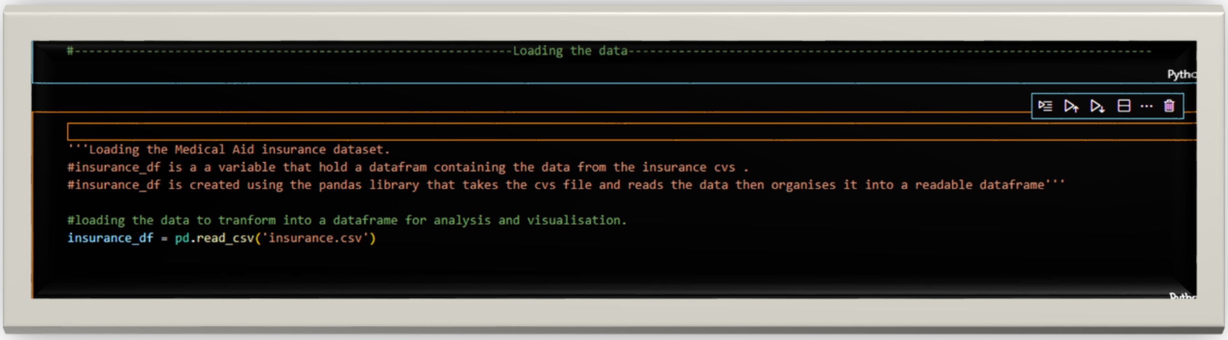
Python libraries can be described as a collection of toolkits with ready-to-use methods to make programming efficient and fast. The libraries assist with programming to not reinvent the wheel for example, for this first application panda's library is used to read a CSV file and the files data into a data frame. Instead of coming up with code to be able to read and put the data into a data frame, the read_csv method is used. The importance of using libraries is explained as, (WsCube Tech, 2025),"A Python library helps us avoid rewriting the same code repeatedly in a program as it contains a collection of codes or modules of codes". Figure 1 shows the different libraries that are used.



```
#------------------------------------------------------------------------Start---------------------------------------------------

#pandas is responsible for data manipulatioon that is why when loading a dataset or reading from it, panda 'pd' is used.
import pandas as pd
#numpy is responsible for arryas and math related programming so when we are doing
import numpy as np
#matplotlib is usef for visuals such as hiistograms describing the data from the dataset so that is why plt is used when plotting graphs.
import matplotlib.pyplot as plt

import chart_studio.plotly as py

import plotly.graph_objects as go
#plotly.colors is used for accessing different colours to use on visuals.
import plotly.colors as pc

#seaborn is also a visualisation toolkit that is used to make visual illustrations. The difference is that seaborn is an extention of matplotlib and uses complex visuals.
import seaborn as sns

#sklearn.model_selction is used for splitting the dataset between training data and testing data during data preparations
from sklearn.model_selection import train_test_split

#sklearn.linear_model import LinearRegression is to access the linear model where the insurance medical data is fitted.
from sklearn.linear_model import LinearRegression

#sklearn.import ElasticNet is the second model to compare accuracy scores with SLP model.
from sklearn.linear_model import ElasticNet

#sklearn library  mean squared_error is used to calculate the mean(average) squared error of the model.
from sklearn.metrics import mean_squared_error

#sklearn.metrics is used to calculate the accuracy score of the model.
from sklearn.model_selection import cross_val_score

#sklearn.model_selection is used to split the dataset into k-folds for cross-validation
```

Figure 1 Python Libraries used

Practical implementation of using python libraries.



```
#-----------------------------------------------------------Loading the data------------------------------------------------------------
```

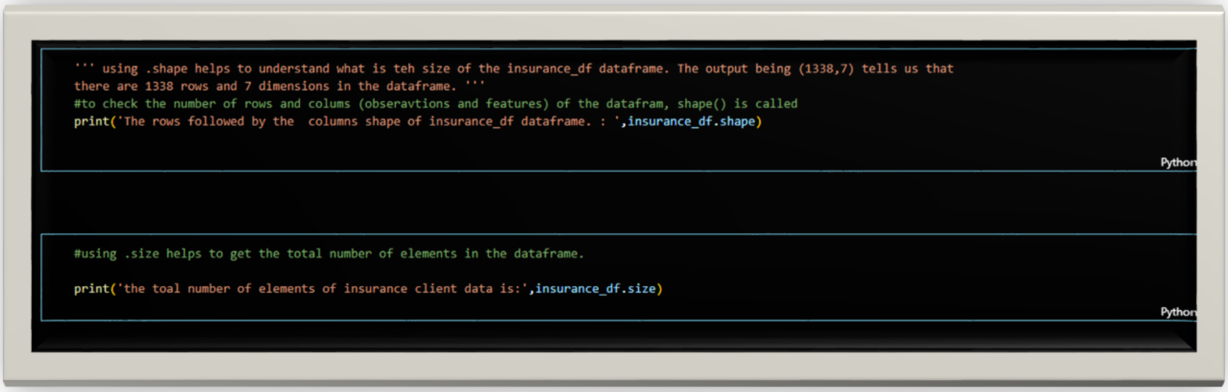
```
'''Loading the Medical Aid insurance dataset.
#insurance_df is a a variable that hold a datafram containing the data from the insurance cvs .
#insurance_df is created using the pandas library that takes the cvs file and reads the data then organises it into a readable dataframe'''

#loading the data to tranform into a dataframe for analysis and visualisation.
insurance_df = pd.read_csv('insurance.csv')
```

Figure 2

Figure two shows using the pandas library that is responsible for data manipulation. When using **pd.read_csv**, the pandas library is used to read the file and put its data into a readable, manipulatable data frame as shown on Figure 2 on page 4.

Exploratory data analysis

When Exploratory data analysis is taking place, this is when the analysis are on step 4. It involves exploring and analysis the distribution, the spread and the central tendency of the insurance client data. To validate this previous statement, it is also mentioned that (GeeksforGeeks, Sanchhaya Education Private Limited, 2025),"The fourth step is to **Analyze**. The cleaned data is used for analyzing and identifying trends"



```
''' using .shape helps to understand what is teh size of the insurance_df dataframe. The output being (1338,7) tells us that
there are 1338 rows and 7 dimensions in the dataframe. '''
#to check the number of rows and colums (obseravtions and features) of the datafram, shape() is called
print('The rows followed by the  columns shape of insurance_df dataframe. : ',insurance_df.shape)
```

```
#using .size helps to get the total number of elements in the dataframe.

print('the toal number of elements of insurance client data is:',insurance_df.size)
```

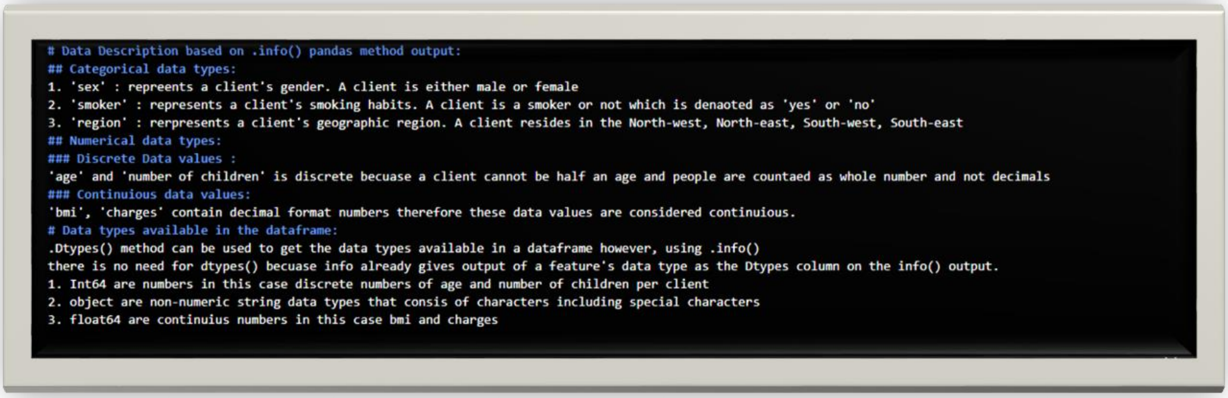
Figure 3 Using pandas library .shape() method

Figure 3 above represents when using the method **.shape()** which provides a summary of the number of rows and columns in the data frame or the data points and dimensions. This summary is useful to get a feel of the size of data being worked on.

Further exploration of the health insurance of the clients

To have a summary of the number of columns, empty or null spaces counted and the data types of the insurance_df data frame, the info() method is used. Figure shows

The datatypes based on the output of the info() method. Figure four shows that there are categorical and numerical variables present. The difference between these variables explained as (Longe, 2025)"Categorical data is also called qualitative data while numerical data is also called quantitative data." And on figure the **categorical variables** are donated as '**object**' data type which includes non-numerical characters, and the **numerical variables** are donated as **'int64' or 'float64'**. Additionally, the insurance_df dataset contains discrete and continuous data which the easiest way to remember the difference is that people cannot be half be in decimal or a person cannot be 23.3 years old for example. therefore data pertaining people, ages and any data that cannot be continuous decimal is observed as discrete variables or data. As shown on on the figure and the code file on page 7, **BMI** and **charges** have a decimal format therefore these are continuous variables. Figure sos and so shows the markdown section explaining the variables including examples.

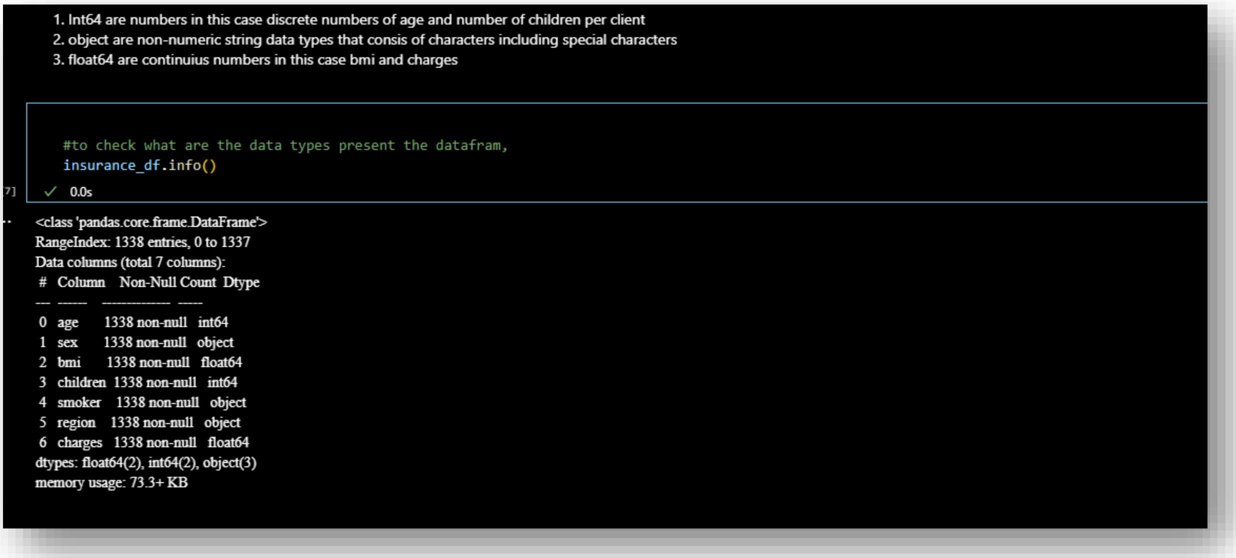


```
# Data Description based on .info() pandas method output:
## Categorical data types:
1. 'sex' : repreents a client's gender. A client is either male or female
2. 'smoker' : represents a client's smoking habits. A client is a smoker or not which is denaoted as 'yes' or 'no'
3. 'region' : rerpresents a client's geographic region. A client resides in the North-west, North-east, South-west, South-east
## Numerical data types:
### Discrete Data values :
'age' and 'number of children' is discrete becuase a client cannot be half an age and people are countaed as whole number and not decimals
### Continiuous data values:
'bmi', 'charges' contain decimal format numbers therefore these data values are considered continiuous.
# Data types available in the dataframe:
.Dtypes() method can be used to get the data types available in a dataframe however, using .info()
there is no need for dtypes() becuase info already gives output of a feature's data type as the Dtypes column on the info() output.
1. Int64 are numbers in this case discrete numbers of age and number of children per client
2. object are non-numeric string data types that consis of characters including special characters
3. float64 are continiuis numbers in this case bmi and charges
```

Figure 4 the different types of variables, data types and understanding what each dimension means

.**head**() helps to return a summary of the first 5 dimensions and observations of the data frame by default. This number can be adjusted. As shown on figure 4

To explain **.dtypes** and .**dtypes.value_count()** return the same information that can be understand by just using .**info**(). The difference is that when using .**dtypes** on its own it only returns what data types are present while **dtypes.value_counts()** retursn how many times a particular data type is presents for example the output of .**dtypes** is shown in on the figure and the code file . It also bring sin ideas of feature engineering and encoding at a glance when spotting which features need to be converted into integers for example.



1. Int64 are numbers in this case discrete numbers of age and number of children per client
2. object are non-numeric string data types that consis of characters including special characters
3. float64 are continiuis numbers in this case bmi and charges

```
#to check what are the data types present the datafram,
insurance_df.info()
```

✓ 0.0s

```
<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)
memory usage: 73.3+ KB
```

This means that 'sex' data is string non numeric, 'BMI' is numeric decimal float64. The output of **.dtypes._value_counts()** as shown on on the figure and the code file which means there are 3 object type features and 2 floats type features and 2 int64 type features. Using Info() gives a clearer sumary of the data frame at once instead of checking for the datatypes individually

On the figure and the code file represent the **output** after using **.tail**() and is similar to .**head**(). The two methods both returns a summary of the number of specified dimensions and observations from the bottom of the data frame for the .tail() and top to bottom when using the .head() method.
Consider 'sex' and 'smoker', for smoker its either yes or no and for sex its either male or female therefore encoding can be perform for these dimensions.
The same would apply for 'region as there are 4 options based on the output of the data frame head.
Encoding will be explored as we understand what type of data we are working and prepare it for the linear regression model.

Using **describe ()**
On the figure and the code file, the number of ages recorded is 1338, the average number of children/dependencies is 1 while the maximum is 5 children/dependencies. The maximum BMI recorded is 53. The summary helps to understand the clients' attributes from this insurance.

Insert null pic here
On the figure and the code file show that there are no null values. The Null values would be caused by examples of when the client does not want to disclose their age or region. No matter what caused the null, it is important to check if the insurance data frame does not contain null values because null values open doors for more inaccuracy. This problem of null values is described as (GeeksforGeeks, Sanchhaya Education Private Limited, 2025),"It's often used as a placeholder for variables that don't hold meaningful data yet. Unlike 0, "", or [], which are actual values,". Considering the direct quote statement having a data frame with unattended null values would affect the model's prediction score because, what would the linear regression model predict if there is no data to train it especially if it is labelled data such as the data being analysed. from the model therefore It is important to check if every client's data is not null.

Analysing the distribution, the spread and central tendency of the insurance data frame feature by feature.

'The following (enter no of cells) are graphs to help check whether:

1. if the clients BMI number affects their insurance chargers
2. if the client's age affects their insurance chargers
3. if the number of children/independents affects their insurance chargers

These correlation checking graphs help to see how this insurance determine their clients' chargers. This will help decide what features to exclude or include, which dimension need encoding'''

1. Age

When analysing the client age distribution, a histogram plot is used to show the distribution of client age. The green histogram on on the figure and the code file shows that there are more youth adult clients than senior adult clients based on the peak of the graph between the ages of 20 -25. The age distribution also suggests that there is a pattern where between 80-85 people are either in their late 20s. 30s, 40s, 50s or 60s

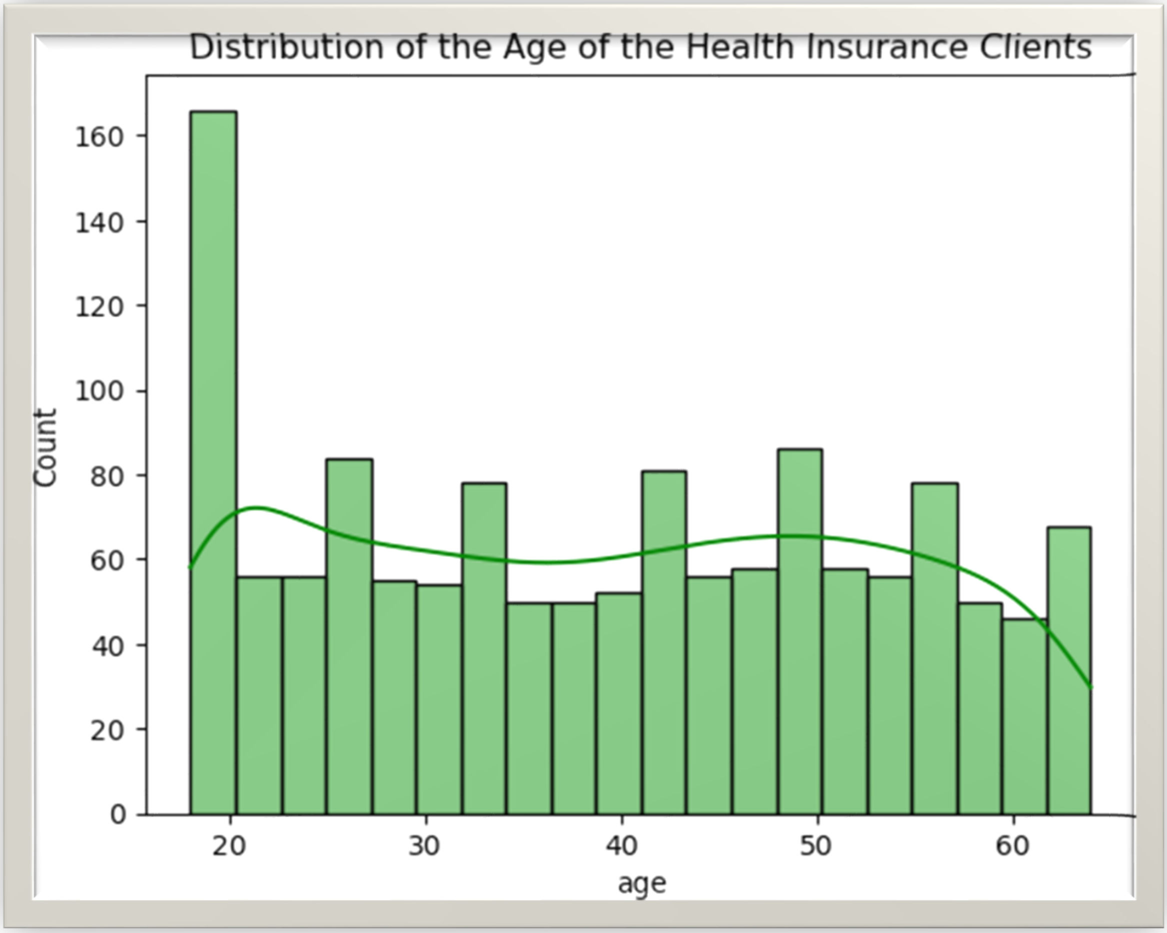
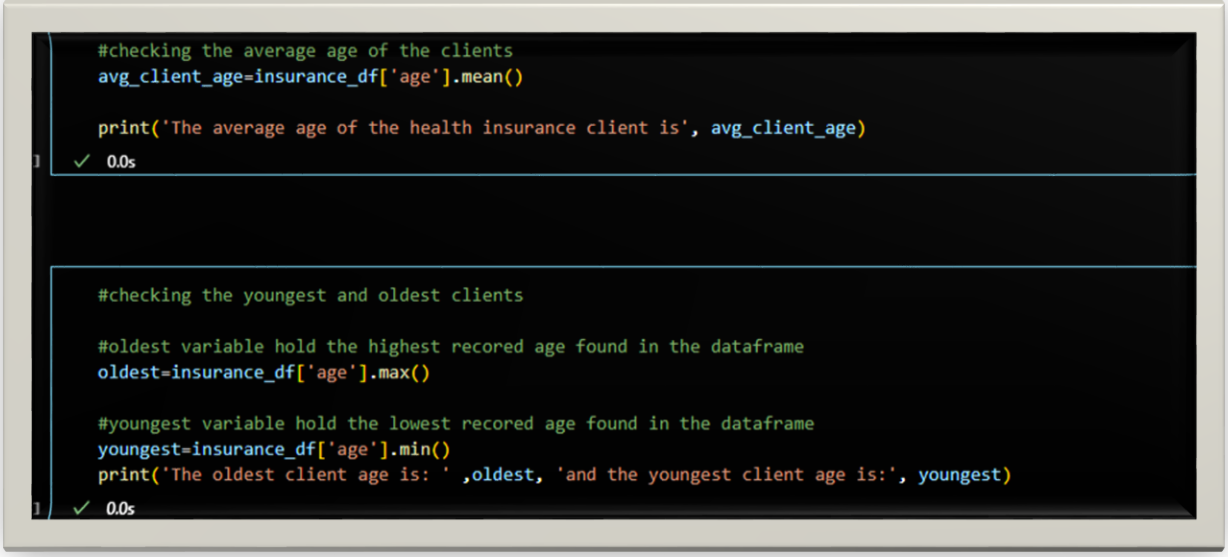


Figure 5 A histogram showing the distribution of the client age in health insurance data frame



```
#checking the average age of the clients
avg_client_age=insurance_df['age'].mean()

print('The average age of the health insurance client is', avg_client_age)
```

✓ 0.0s

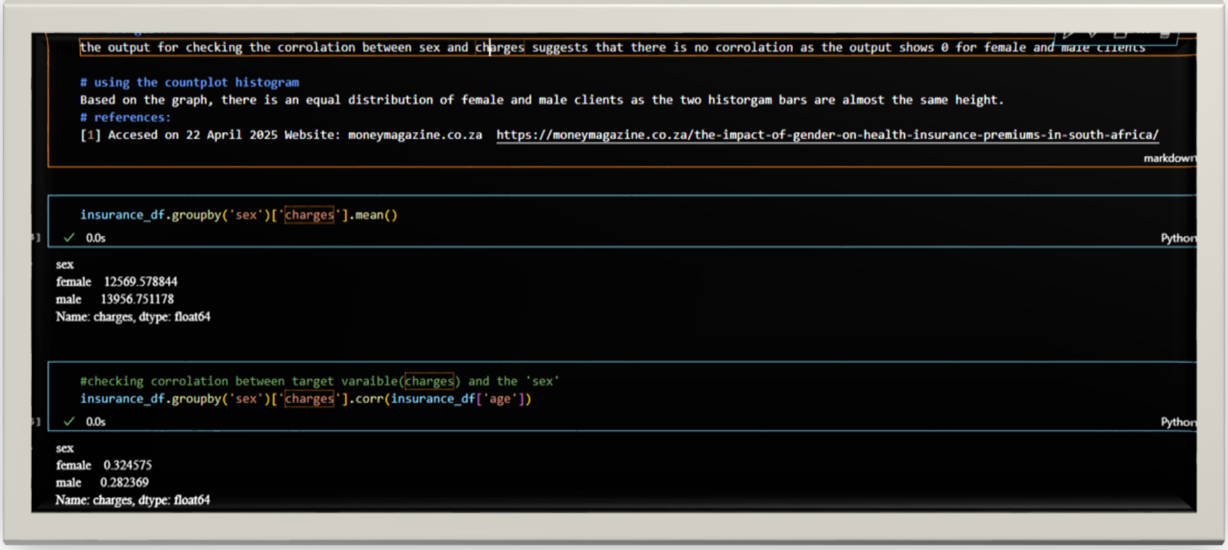
```
#checking the youngest and oldest clients

#oldest variable hold the highest recored age found in the dataframe
oldest=insurance_df['age'].max()

#youngest variable hold the lowest recored age found in the dataframe
youngest=insurance_df['age'].min()
print('The oldest client age is: ' ,oldest, 'and the youngest client age is:', youngest)
```

✓ 0.0s

Figure 6 using pandas to retrieve data to calculate average age, oldest age and youngest age

On the figure and the code file show how the average of the client has been calculated with the help of the python library pandas instead of manually coding a method for example to calculate the mean. This is an advantage of using python library as it serves as a toolkit to make data analysis effective.

Gender



```
the output for checking the corrolation between sex and charges suggests that there is no corrolation as the output shows 0 for female and male clients

# using the countplot histogram
Based on the graph, there is an equal distribution of female and male clients as the two historgam bars are almost the same height.
# references:
[1] Accesed on 22 April 2025 Website: moneymagazine.co.za  https://moneymagazine.co.za/the-impact-of-gender-on-health-insurance-premiums-in-south-africa/
```

```
insurance_df.groupby('sex')['charges'].mean()
```

```
sex
female    12569.578844
male      13956.751178
Name: charges, dtype: float64
```

```
#checking corrolation between target varaible(charges) and the 'sex'
insurance_df.groupby('sex')['charges'].corr(insurance_df['age'])
```

```
sex
female    0.324575
male      0.282369
Name: charges, dtype: float64
```

Figure 7 showing the code snippet checking if there is association with gender and charges variables

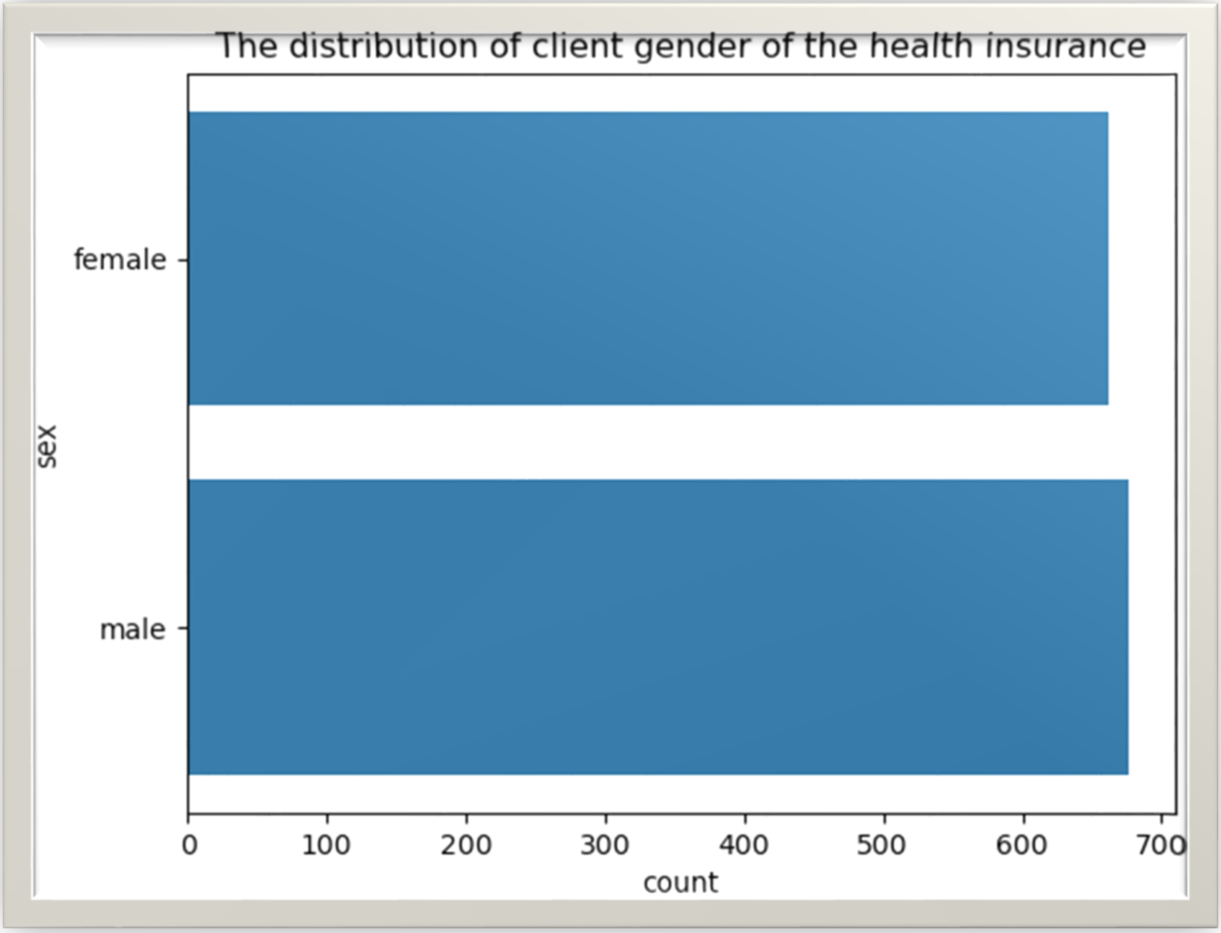


Figure 8 A count plot graph showing the count of gender (Female and Male) clients

According to an article, there are factors that influence the average charges between male and female clients are (Money Magazine,2023)"Historically, women have been found to require more frequent medical care, including reproductive health services such as prenatal care, family planning, and maternity-related expenses." so clients might pay an higher average based on the medical goods and services they require for example the female clients insurance might have additional benefits for woman health care. Consider the correlation output of gender and charges variables, it shown a 0.3 for females and 0.2 for male clients. There is no association relationship between gender and the charges clients pay.

BMI

Looking at the red histogram and blue boxplot, the outliers for BMI are between 45 and 50 while the rest if the data points for BMI are between 25 and 35, and the average BMI recorded is 30. The average is 30 as it was confirmed using the .mean() method that return the average value.

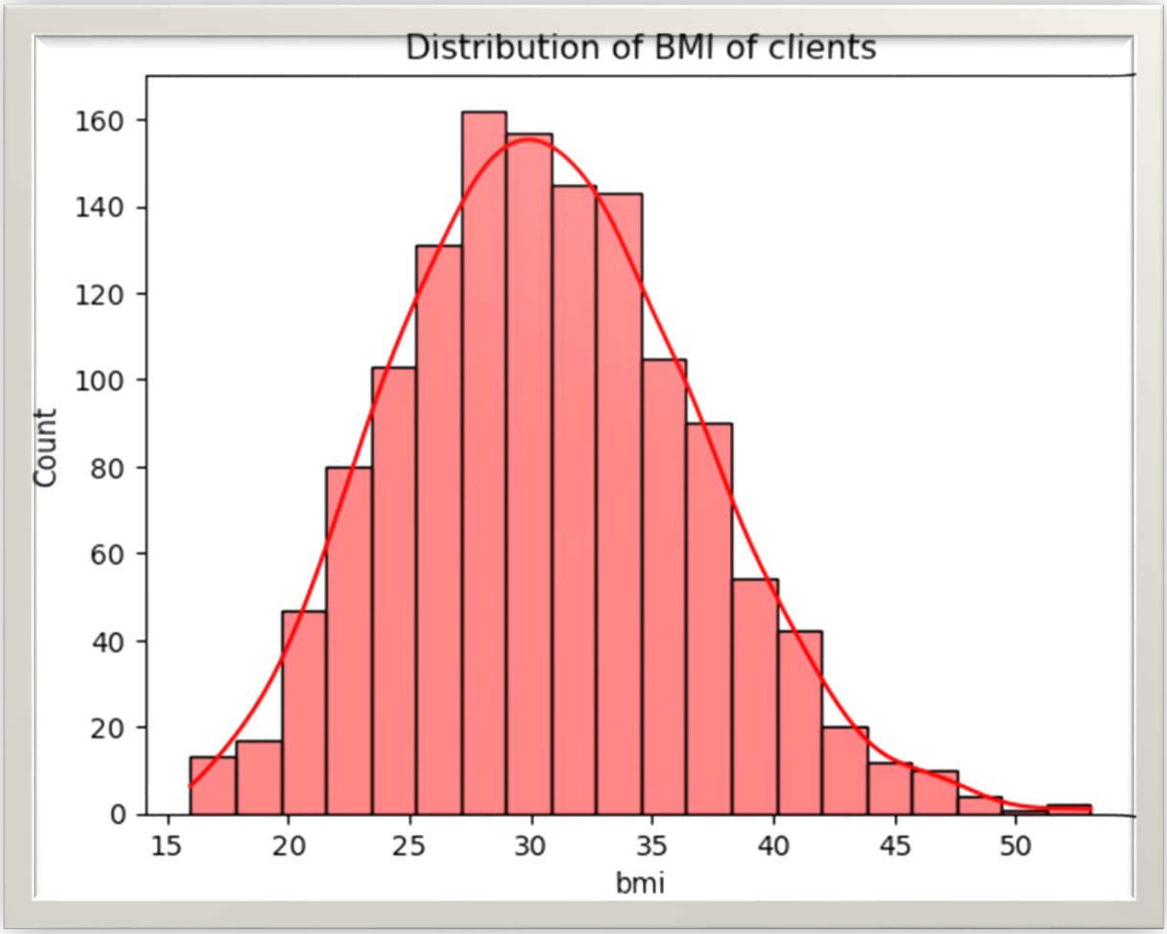


Figure 9 A histogram showing the distribution of the client's Body Mass Index measurements

Figure 9 shows us that there are more clients with a BMI between 25 and 40. The graphs shows that there are over 140 clients with a BMI of 30 and less than 20 clients with a BMI of 45. The outliers are the clients with a BMI between 15 and 20 or a BMI between 45 and 50.
The graph also suggests that a client's Body Mass Index number has an influence on health insurance needs. Consider the boxplot:

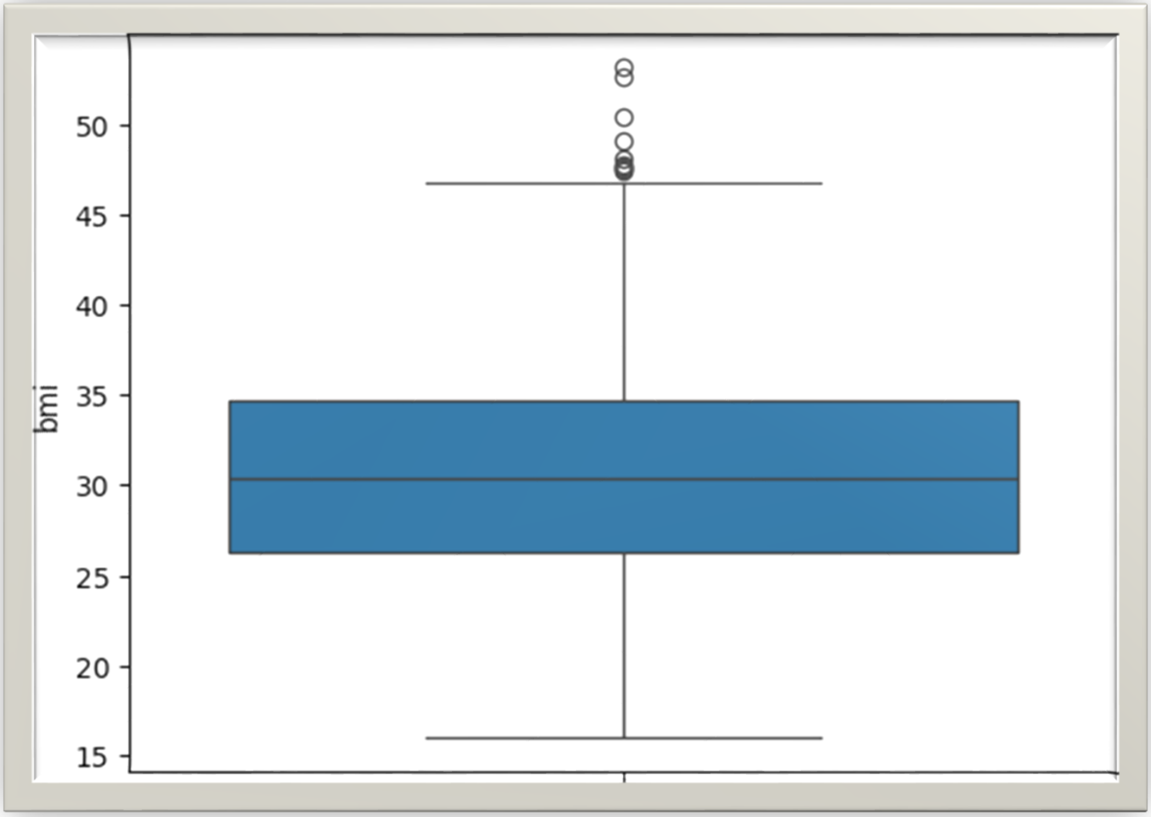


Figure 10 a boxplot showing the average of 30 BMI denoted as the middle horizontal line

The number of children data distribution

Another type of plot is introduced is also created using seaborn library, where all features data are compared to but this is for numerical data. Looking at bottom right last grid on figure 11, it shows the effect of the number of children/dependants it has on a client's charges. This suggests that the more people who are part of a client's beneficiary list, adds more charges. The less children/dependants a client has, the less charges the clients pay for health insurance.

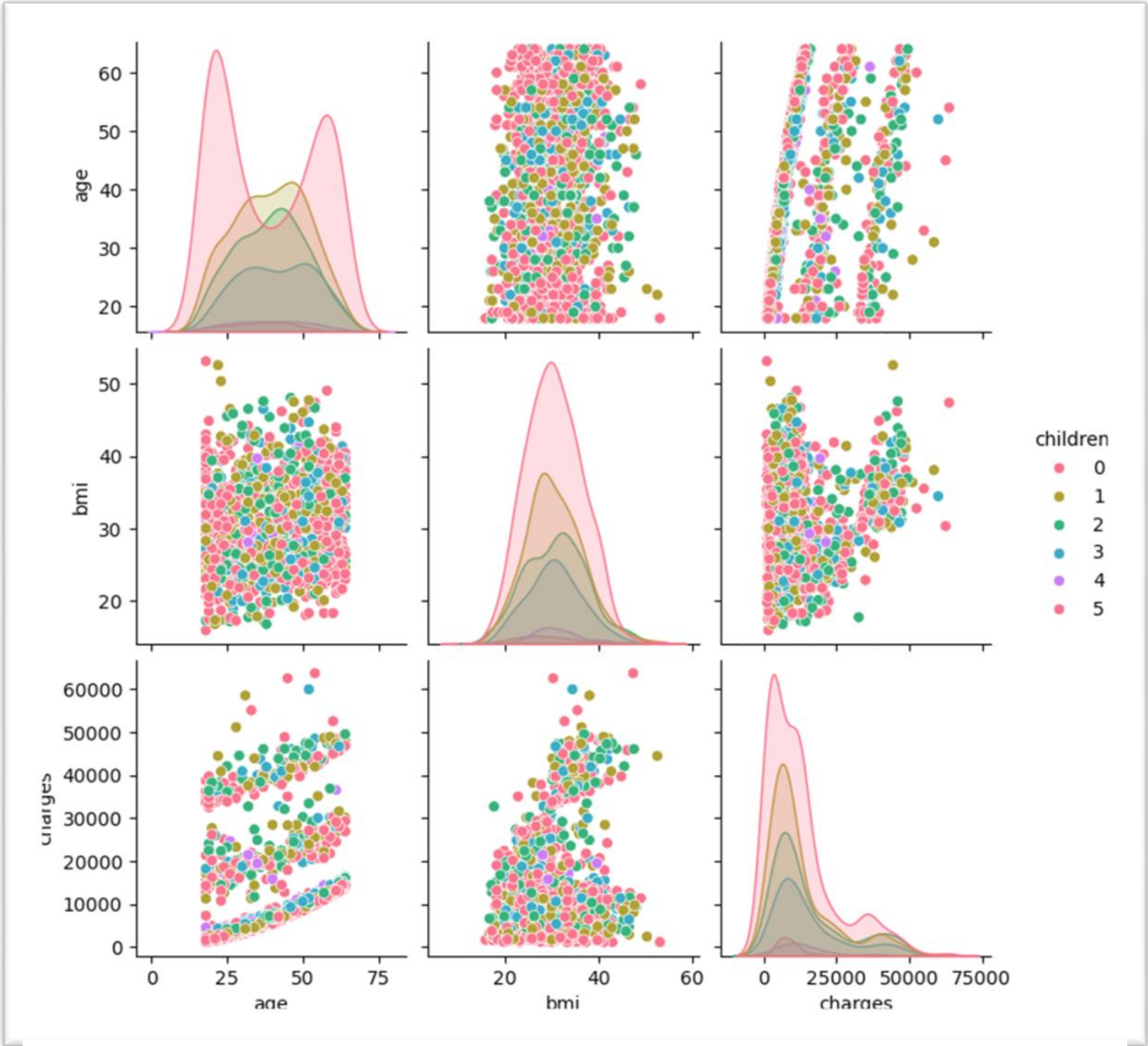


Figure 11 a pair plot representing numeric client data compared to the other variables in the data frame

Client Smoking habits data distribution analysis

Consider the graph below:

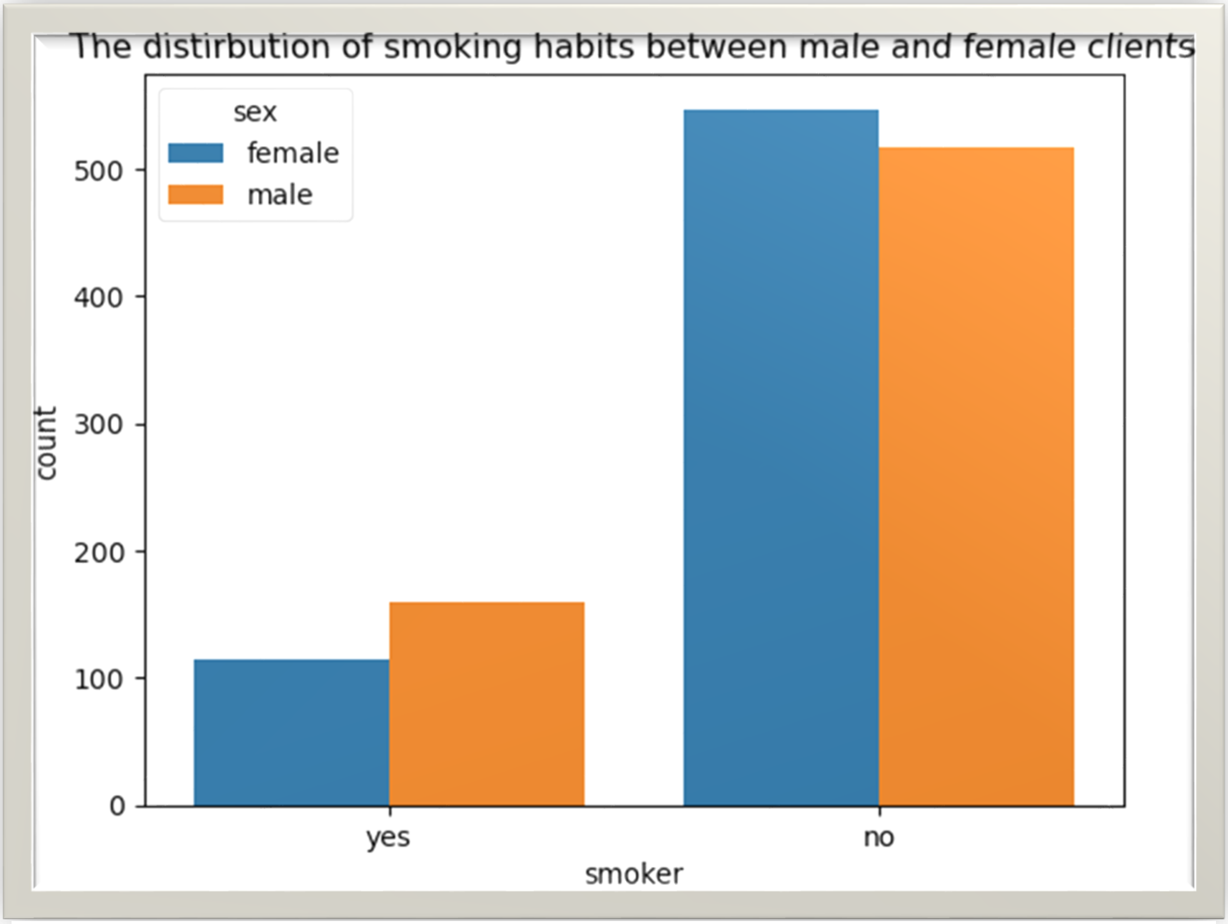
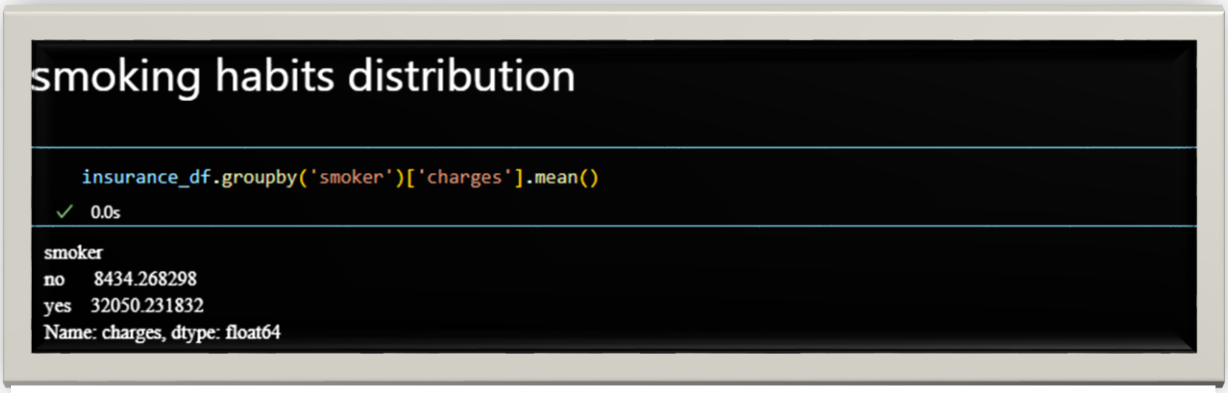


Figure 12 A histography representing the count of female smoker and non-smoker and male smoker and non-smoker

Figure 12 and 13 show the average of chargers based on the clients' smoking habits.



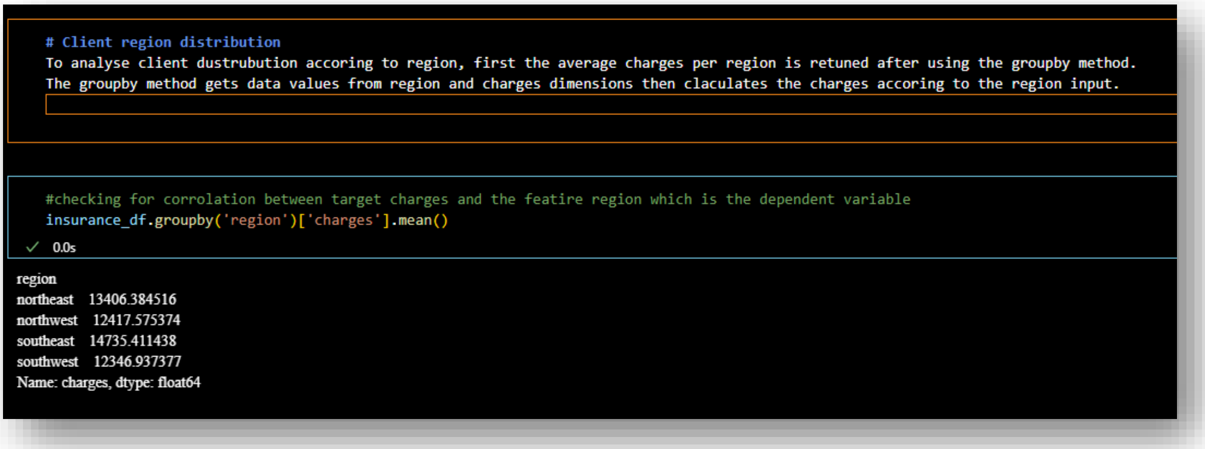
smoking habits distribution

```
insurance_df.groupby('smoker')['charges'].mean()
✓ 0.0s

smoker
no      8434.268298
yes    32050.231832
Name: charges, dtype: float64
```

Figure 13

Client Region distribution

The average charges paid by client that live in the northwest and southwest region (NW=12 417, SW=12 346) pay less charges compared to northeast and southeast client regions. This suggests the difference in the standard of the east region is higher than of the west and/or clients from the eastern regions choose higher charges because of the area they live in. Maybe there are a more chances of getting injured and require medical assistance compared to living in the western region where there are less chances of getting injured and require health emergency assistance.



```
# Client region distribution
To analyse client dustrubution accoring to region, first the average charges per region is retuned after using the groupby method.
The groupby method gets data values from region and charges dimensions then claculates the charges accoring to the region input.
```

```
#checking for corrolation between target charges and the featire region which is the dependent variable
insurance_df.groupby('region')['charges'].mean()
```

```
region
northeast    13406.384516
northwest    12417.575374
southeast    14735.411438
southwest    12346.937377
Name: charges, dtype: float64
```

Consider the boxplot below:

Region shows to have an effect in the amount of charges a client pays based on the output of the boxplot and **grouby()** method. There are outliers as shown on the boxplot that deviate away from the average client region health insurance charges

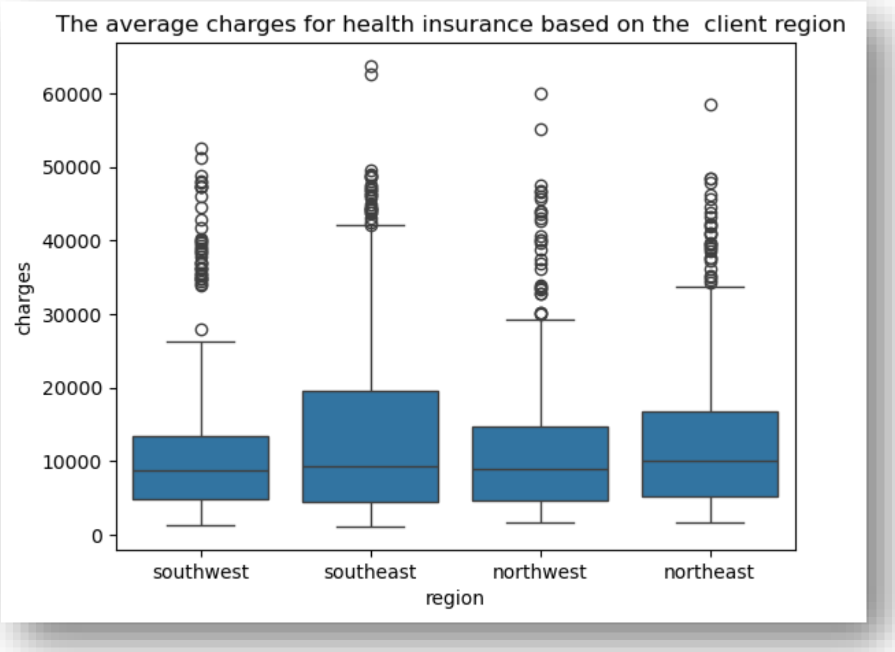


Figure 14 a boxplot representing the average charges for health insurance clients based on their residential region

Client Chargers distribution based on gender

The boxplot below here on page 19 shows a boxplot show the average charges between female and male. Both male and female clients have a mean line draw over its box plot between 9000 – 10 000 United States Dollars. The boxplot also gives insight of this U.S health insurance client data as that the health insurance does not change their charges based on gender as both features have equal averages calculated. There is an equal distribution of charges based on

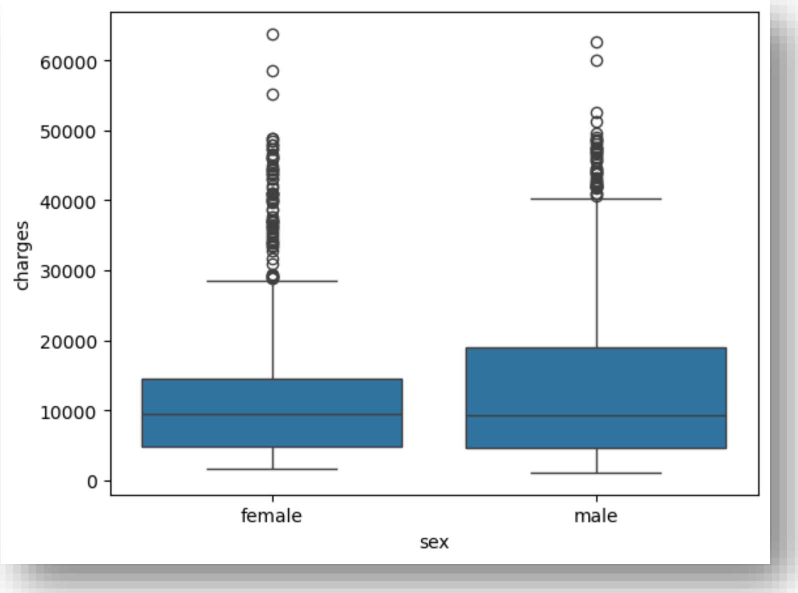


Figure 15 a boxplot representing the equal average charges between female clients and male clients

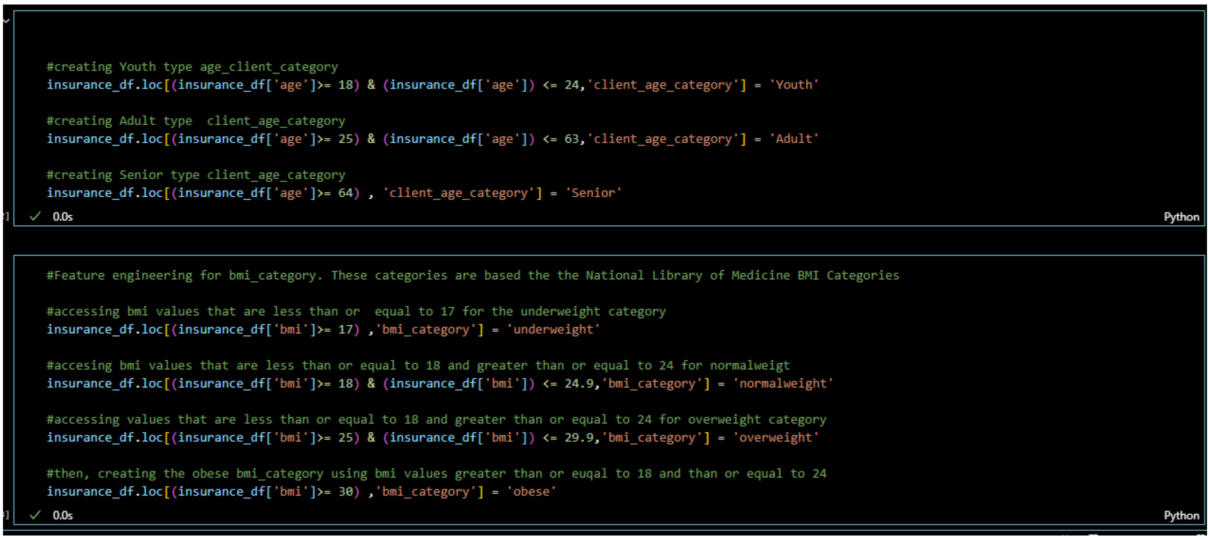
Feature engineering: What is it and why is it important and used?
Looking at age and **bmi features,** feature engineering is handy when wanting to create new features such as **age_category** where this feature is like smoker with a yes or no, for **age_category** the plan is to create 3 types of age clients (youth, adult and senior) and for bmi the plan is to create a new feature called **bmi_category** with 3 types of bmi of a client that is underweight, normal, overweight and obese. According to Healthline.com (Cirino, 2016),"BMI Weight Status Below 18.5 Underweight 18.5 – 24.9 Normal 25.0 – 29.9 Overweight 30.0 and above Obese" therefore a client's bmi data value determines what category they fall under."

The bmi category is based on the US document that are available to access online. please refer to the reference list on the last page. After feature engineering, these categories are encoded using one hot encoding.

Encoding: What it is and why is it important for linear regression model

On Data Camp’s organisational website they say (DataCamp, 2025),"One-hot encoding is a technique used to convert categorical data into a binary format where each category is represented by a separate column with a 1 indicating its presence and 0s for all other categories." so looking at figure 17 on page 22, after feature engineering, it is possible to encode the age_category, bmi_category, sex, smoking habits, region.

The reason for encoding so many features is because with linear regression, it is best to work with numeric data values instead of taking extra steps prone to error by using the categorical variables straight. To clarify the previous statement, consider this point written on Saturn Cloud organisational website (SaturnCloud, 2025),"[w]hen we use linear regression to model the relationship between a dependent variable and one or more independent variables, we assume that the independent variables are continuous and can take on any value.”



```
#creating Youth type age_client_category
insurance_df.loc[(insurance_df['age']>= 18) & (insurance_df['age']) <= 24,'client_age_category'] = 'Youth'

#creating Adult type  client_age_category
insurance_df.loc[(insurance_df['age']>= 25) & (insurance_df['age']) <= 63,'client_age_category'] = 'Adult'

#creating Senior type client_age_category
insurance_df.loc[(insurance_df['age']>= 64) , 'client_age_category'] = 'Senior'
```

✓ 0.0s Python

```
#Feature engineering for bmi_category. These categories are based the the National Library of Medicine BMI Categories

#accessing bmi values that are less than or  equal to 17 for the underweight category
insurance_df.loc[(insurance_df['bmi']>= 17) ,'bmi_category'] = 'underweight'

#accesing bmi values that are less than or equal to 18 and greater than or equal to 24 for normalweigt
insurance_df.loc[(insurance_df['bmi']>= 18) & (insurance_df['bmi']) <= 24.9,'bmi_category'] = 'normalweight'

#accessing values that are less than or equal to 18 and greater than or equal to 24 for overweight category
insurance_df.loc[(insurance_df['bmi']>= 25) & (insurance_df['bmi']) <= 29.9,'bmi_category'] = 'overweight'

#then, creating the obese bmi_category using bmi values greater than or euqal to 18 and than or equal to 24
insurance_df.loc[(insurance_df['bmi']>= 30) ,'bmi_category'] = 'obese'
```

✓ 0.0s Python

Figure 16 code snippet showing feature engineering on insurance_df

Without encoding the new features which are categorical variables, it would add additional steps because the linear regression model works out the outcome based on the relationship between the dependant and independent variables in our case our independent variable is charges but the other variables without encoding are not numeric, also the model is expecting the variables to be numeric(continuous or discrete) so encoding is a requirement as we are building a regression model.

The age categories are based on the spread of client age in the insurance_df data frame. Youth is capped from 18- 24 because based on the output of the count plot histogram, the graph peaked after counting between 20–30-year-old clients which are approximately 60-64 years old.

Adult is capped between 25-63 because creating a range that are 10 years apart which mean 1 age range is 20-29, the other is 30-39 etc until we reach the oldest age which is 64.Doing it this way will add up time for processing power therefore Adult category age range is between 30-59 and senior client age range is 60 years old and above

Additional reasons, consider this statement by the South African Department of Public Service and Administration (2019),"In terms of the Public Service Act, 1994 (PSA), the normal retirement age of employees is 65 years, and such employees will, on retirement, retire with no pension penalties" so that means the expected retirement age is approximately 60 years old in South Africa so considering a client that is 45 as senior would not make sense, Using the data in the south African context as the data analysis process steps into fitting the data, because it is required the model helps the South African medical aid scheme gain insight and solve heir current problem of not having tailored charges for their clients. Making the senior category feature be capped at 64 and older make sense. This would not make sense because if the model's outcomes are interpreted by the south African medical aid scheme, they might conclude their business decision as 'youth clients pay less medical aid charges because they are not a 49-year-old senior'

Discussing the correlation heatmap

(Wagavkar & Whitfield , 2024),"Most data scientists consider the use of a correlation matrix as the main step before building any machine learning model because if you know which variables are correlated which, you can gain a better understanding about what's most important for your model." where the above cell uses the Corr() then plots it into a correlation heatmap to show colour gradient variation between two variables based on how associated they are to each other and Wamankar adds (2024),"We can use a correlation matrix to summarize a large data set and to identify patterns and make a decision according to it." Based on the correlation heatmap output this has been analysed:

If charges is the target variable(y) and age, BMI, etc are the feature, we can interpret this correlations as:

- smoking habits which is represented as smoker_yes and smoker_no after encoding and charges, the correlation matrix lies at 0.79 and the smoker_no feature is at -0.79 BMI and charges correlations matrix is 0.2 where bmi_category_obese is 0.2 and bmi_category_overweight is -0.2

age and charges correlation matrix is 0.3. Positive matrixes indicate that there is a positive correlation while negative matrixes indicate there is a negative correlation hence:

- age and charges has a positive correlation, smoking habits f and charges has a positive correlation. This suggests that the more a client smokes, the more they need medical health care services therefore their charges increase.

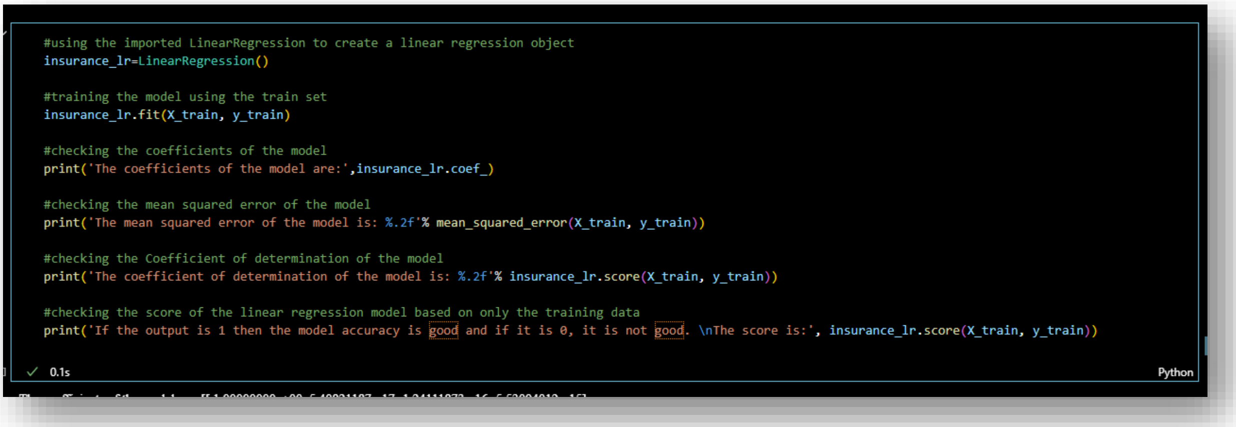
- BMI and charges has a positive correlation therefore the closest a client falls under BMI category overweight; the health insurance charges increase. This suggests that clients might use medial aid to either move to a lower BMI or higher therefore the client requires more health insurance to cover all the bills.

- Age and charges has a positive correlation and suggests that the older a client , the more medical services they require and more charges they need to pay to cover hospital bills for example.

Conclusion based on Linear Regression Models outcomes

The following figures 16 through 17 below, show the implementation of the Simple Linear Regression Model:

After splitting the training from the testing data , the first model which is the Simple Linear Regression model has been used.



```
#using the imported LinearRegression to create a linear regression object
insurance_lr=LinearRegression()

#training the model using the train set
insurance_lr.fit(X_train, y_train)

#checking the coefficients of the model
print('The coefficients of the model are:',insurance_lr.coef_)

#checking the mean squared error of the model
print('The mean squared error of the model is: %.2f'% mean_squared_error(X_train, y_train))

#checking the Coefficient of determination of the model
print('The coefficient of determination of the model is: %.2f'% insurance_lr.score(X_train, y_train))

#checking the score of the linear regression model based on only the training data
print('If the output is 1 then the model accuracy is good and if it is 0, it is not good. \nThe score is:', insurance_lr.score(X_train, y_train))
```

Figure 17 Code snippet of training the model

The output after fitting the train set(X_train and y_train):

what these coefficient mean?

The coefficients of the model are: [[1.00000000e+00 5.40821187e-17 -1.24111873e-16 -5.53094012e-15]

[4.00662052e-14 1.00000000e+00 -6.23554468e-14 -2.22044605e-16]

[1.36045697e-17 -4.59752543e-17 1.00000000e+00 -9.55114351e-18]

[1.48383098e-13 -8.15791878e-13 3.87620146e-13 1.00000000e+00]]

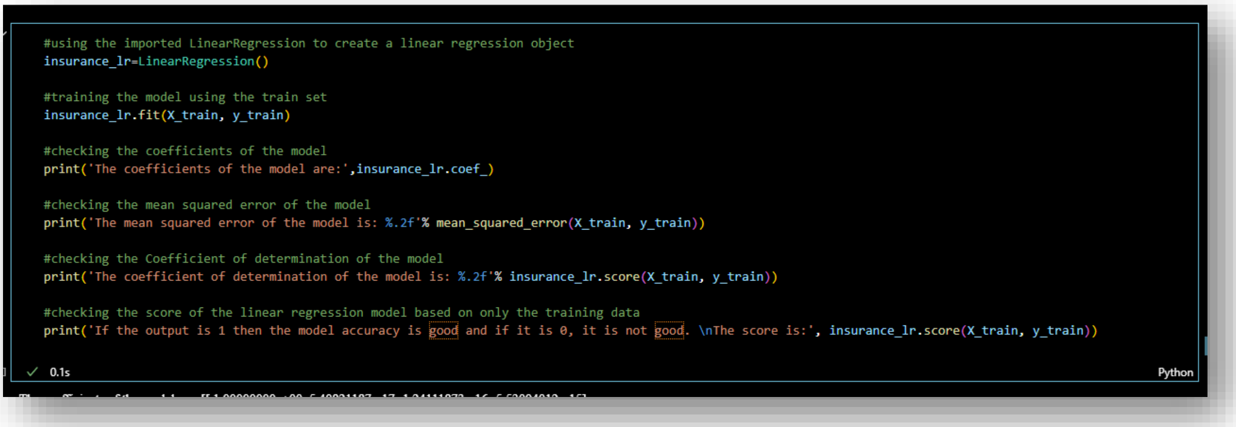
what the mean squared error output mean?

The mean squared error of the model is: 0.00

what the coefficient of determination means?

The result is 1.00 when converted into a percentage it is 100%. This suggests that the model is overfitting where there is a high bias and a low variance.

Another possibility is to perform folding. The guide explains this as (Polamuri , 2024),"A technique where our data is split into multiple folds and the model is trained and tested multiple times" and based on this statement it is concluded to use cross validation is required to ensure that the reliability of this model is consistent throughout in different scenarios. Testing the model followed and its showed after figure 17 below:



```
#using the imported LinearRegression to create a linear regression object
insurance_lr=LinearRegression()

#training the model using the train set
insurance_lr.fit(X_train, y_train)

#checking the coefficients of the model
print('The coefficients of the model are:',insurance_lr.coef_)

#checking the mean squared error of the model
print('The mean squared error of the model is: %.2f'% mean_squared_error(X_train, y_train))

#checking the Coefficient of determination of the model
print('The coefficient of determination of the model is: %.2f'% insurance_lr.score(X_train, y_train))

#checking the score of the linear regression model based on only the training data
print('If the output is 1 then the model accuracy is good and if it is 0, it is not good. \nThe score is:', insurance_lr.score(X_train, y_train))
```

Figure 18 Code snippet showing testing the model

The line after testing the data by fitting it into the SLR model evaluation

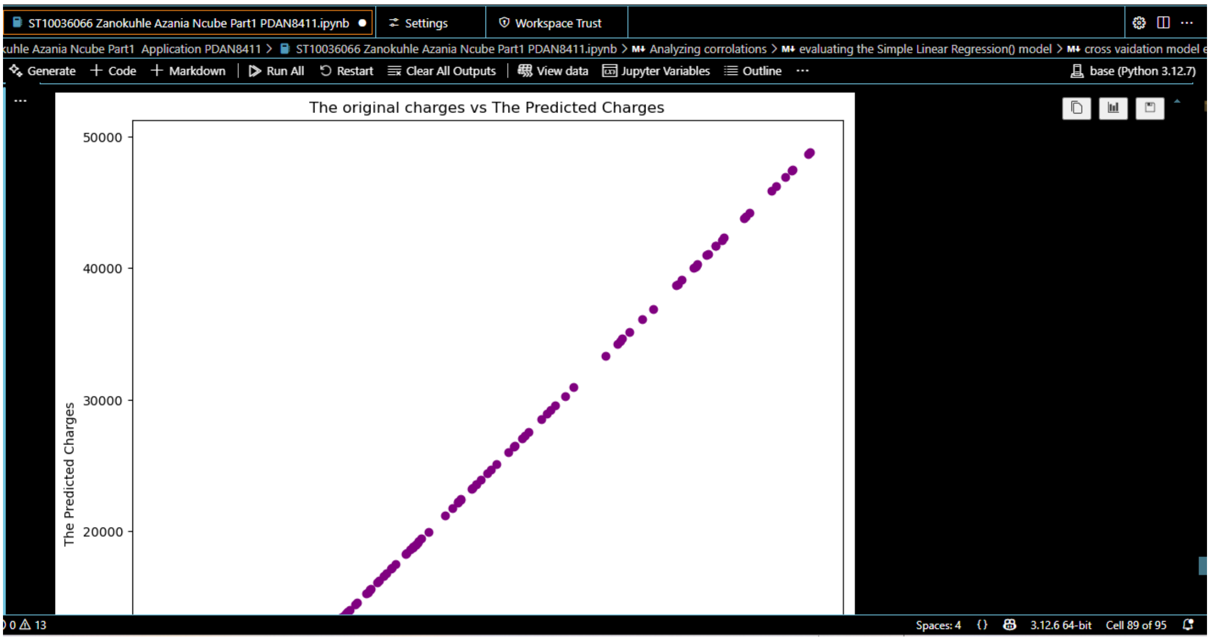


Figure 19 linear regression line (The original charges vs the predicted charges)

What is ElasticNet ()

The second model used is the ElasticNet() model which a combination of L1 regularisation(Lasso) and L2 Regularisation (Ridge). Duhmne explains regularisation as (Dhumne, 2023),"It is a regularized regression technique that is used to deal with the problems of multicollinearity and overfitting, which are common in high-dimensional datasets."

Dhume further explains the L1 and L2 regularisation responsibilities as (2023)"The L1 norm is used to perform feature selection, whereas the L2 norm is used to perform feature shrinkage." therefore instead of using the Lasso() and Ridege() separately, ElasticNet() was the closest model to the two.

Interpreting the output after fitting the data into the ElasticNet () model:

The printed output resulted as:

```
The coefficients of the model are: [[ 9.97216674e-01  0.00000000e+00   0.00000000e+00  9.11839835e-07]
 [ 0.00000000e+00   9.86282195e-01 -0.00000000e+00  1.41695767e-06]
 [ 0.00000000e+00  -0.00000000e+00  7.02188865e-01  2.57335774e-06]
 [ 0.00000000e+00   0.00000000e+00  0.00000000e+00  9.99999997e-01]]
The mean squared error of the model is: 0.03
```

If the output is 1 then the model accuracy is good and if it is 0, it is not good 0.9775844794573616. As shown in the output, this time the score is zero and recalling after using the simple linear Regression model its score is 1.0 more than 1 after folding and cross-validation .

Conclusion

This difference suggest that using the Simple Linear Regression model , the prediction strength of the linear model is stronger than the ElasticNet(). The scatterplot also display the same difference where the simple linear regression shows a straight purple line representing the original data (test) and the trained data. This is the reason why the testing set had to be left alone so that models do not over perform.In the next part, feedback on part 1 is implemented and the model is need improvement based on new requirements by the South African Medical Aid scheme organisation.

References

Wagavkar , S. & Whitfield , B., 2024. *Introduction to the Correlation Matrix builtin.com.* [Online] Available at: <https://builtin.com/data-science/correlation-matrix Introduction to the Correlation Matrix [accessed 25 April 2025] 2024>
[Accessed 25 April 2025].

Choi, M., 2018. *Medical Cost Personal Datasets:Insurance Forecast by using Linear Regression.* [Online] Available at: https://www.kaggle.com/datasets/mirichoi0218/insurance?resource=download [Accessed 14 April 2025].

Cirino, E., 2016. *Body Mass Index for Adults: Healthline.* [Online] Available at: https://www.healthline.com/health/body-mass-index#Body-Mass-Index-for-Adults [Accessed 23 April 2025].

DataCamp, 2025. *DataCamp,2025.One Hot-Encoding Tutorial.* [Online] Available at: https://www.datacamp.com/tutorial/one-hot-encoding-python-tutorial [Accessed 23 April 2025].

GeeksforGeeks, Sanchhaya Education Private Limited, 2025. *Null in python.* [Online] Available at: https://www.geeksforgeeks.org/null-in-python/ [Accessed 25 April 2025].

GeeksforGeeks, Sanchhaya Education Private Limited, 2025. *Six Steps of Data Analysis Process. GeekforGeeks.* [Online]
Available at: https://www.geeksforgeeks.org/six-steps-of-data-analysis-process/ [Accessed 25 April 2025].

Longe, B., 2025. *Categorical vs Numerical Data: 15 Key Differences & Similarities. FormPl.* [Online] Available at: https://www.formpl.us/blog/categorical-numerical-data [Accessed 26 April 2025].

Polamuri , S., 2024. *An in-depth guide to linear regression.* [Online] Available at: https://dataaspirant.com/linear-regression/#t-1697350902759 [Accessed 25 April April].

Rilityane, N., 2019. *Early retirrement wihtout penalisation of pension benefits in terms of section 16(6) of the public act ,1994 - Erratum.* [Online] Available at: https://www.dpsa.gov.za/dpsa2g/documents/cos/2019/ERWRPB_18_06_2019.pdf [Accessed 25 April 2025].

SaturnCloud, 2025. *Linear regression with sklearn using categorical variables. SaturnCloud.* [Online] Available at: :<https://saturncloud.io/blog/linear-regression-with-sklearn-using-categorical-variables/#:~:text=To%20use%20categorical%20variables%20in,label%20encoding%2C%20and%20binary%20encoding
[Accessed 25 April 2025].

Weir, C. B. & Jan., A., 2025. *BMI Classification Percentile and Cut Off Points. National Library of Medicine.* [Online] Available at: https://www.ncbi.nlm.nih.gov/books/NBK541070/ [Accessed 24 April 2025].

WsCube Tech, 2025. *Python Libraries: Top Lists, Uses, How to Choose.* [Online] Available at: https://www.wscubetech.com/resources/python/libraries [Accessed 15 April 2025].