**Sample ChatGPT Thread - Data Science**

In this exercise we will try to build a Linear Regression model with the help of ChatGPT. Let’s get started:

**Key:**

* **Query to ChatGPT**
* **ChatGPT’s Response**
* **Analysis of Response**

**Query:**

I want to build an OLS model using statsmodels library and the dietox dataset. Can you generate the code for me?

**Response:**

Certainly! I can help you generate the code for building an Ordinary Least Squares (OLS) model using the statsmodels library and the dietox dataset. Here's an example code snippet to get you started:

import pandas as pd

import statsmodels.api as sm

# Load the dietox dataset

data = sm.datasets.get\_rdataset('dietox').data

# Create a DataFrame with the relevant variables for the model

df = pd.DataFrame(data, columns=['Weight', 'Feed', 'Time'])

# Add a constant column to the DataFrame

df['Intercept'] = 1

# Define the dependent variable

y = df['Weight']

# Define the independent variables

X = df[['Intercept', 'Feed', 'Time']]

# Create the OLS model

model = sm.OLS(y, X)

# Fit the model to the data

results = model.fit()

# Print the summary statistics of the model

print(results.summary())



***Analysis of the response:***

*The response looks reasonable and complete. However, when I take the code and try to run on my machine, I get an error:*

*URLError: <urlopen error [SSL: CERTIFICATE\_VERIFY\_FAILED] certificate verify failed: unable to get local issuer certificate (\_ssl.c:1125)>*

*So, I ask ChatGPT to help me resolve this issue.*

**Next Query to ChatGPT:**

When I run the code I am getting the following error:  
  
URLError: <urlopen error [SSL: CERTIFICATE\_VERIFY\_FAILED] certificate verify failed: unable to get local issuer certificate (\_ssl.c:1125)>  
  
How do I resolve it?

**Response:**

The error you encountered, URLError: <urlopen error [SSL: CERTIFICATE\_VERIFY\_FAILED] certificate verify failed: unable to get local issuer certificate (\_ssl.c:1125)>, typically occurs when there is an issue with the SSL certificate verification during the download of the dataset using get\_rdataset function.

To resolve this issue, you can try the following steps:

Step 1: Install the certifi package

pip install certifi



Step 2: Import and set the SSL certificate verification using certifi in your code before loading the dataset.

import certifi

import ssl

ssl.\_create\_default\_https\_context = ssl.\_create\_default\_https\_context(certfile=certifi.where())

This code sets the SSL certificate verification using the certifi package, which provides a bundle of CA certificates that can be used for verification.

Here's the updated code with the SSL certificate verification:

import pandas as pd

import statsmodels.api as sm

import certifi

import ssl

ssl.\_create\_default\_https\_context = ssl.\_create\_default\_https\_context(certfile=certifi.where())

# Load the dietox dataset

data = sm.datasets.get\_rdataset('dietox').data

# Create a DataFrame with the relevant variables for the model

df = pd.DataFrame(data, columns=['Weight', 'Feed', 'Time'])

# Add a constant column to the DataFrame

df['Intercept'] = 1

# Define the dependent variable

y = df['Weight']

# Define the independent variables

X = df[['Intercept', 'Feed', 'Time']]

# Create the OLS model

model = sm.OLS(y, X)

# Fit the model to the data

results = model.fit()

# Print the summary statistics of the model

print(results.summary())

By setting the SSL certificate verification using certifi, you should be able to resolve the URLError related to the SSL certificate.

***Analysis of the response:***

*The new code gave me a new error:*

*TypeError: create\_default\_context() got an unexpected keyword argument 'certfile'*

*So, I ask ChatGPT to fix it.*

**Next Query to ChatGPT:**

Now I am getting a new error:

TypeError: create\_default\_context() got an unexpected keyword argument 'certfile'

How do I fix it?

**Response:**

I apologize for the confusion. It appears that the certifi package does not support the certfile argument in the create\_default\_context() function, which caused the new error.

To resolve this issue, you can use a different approach to disable SSL certificate verification for downloading the dataset. Here's an updated code snippet:

import pandas as pd

import statsmodels.api as sm

import ssl

import urllib.request

# Disable SSL certificate verification

ssl.\_create\_default\_https\_context = ssl.\_create\_unverified\_context

# Load the dietox dataset

url = 'https://vincentarelbundock.github.io/Rdatasets/csv/boot/dietox.csv'

data = pd.read\_csv(url)

# Create a DataFrame with the relevant variables for the model

df = pd.DataFrame(data, columns=['weight6', 'feed', 'time'])

# Add a constant column to the DataFrame

df['Intercept'] = 1

# Define the dependent variable

y = df['weight6']

# Define the independent variables

X = df[['Intercept', 'feed', 'time']]

# Create the OLS model

model = sm.OLS(y, X)

# Fit the model to the data

results = model.fit()

# Print the summary statistics of the model

print(results.summary())



***Analysis of the response:***

*Now I get a new error:  
  
HTTPError: HTTP Error 404: Not Found*

*Let us ask ChatGPT to fix this problem. Before that pay attention to the ‘weight6’ column/variable in the code. We will come back to this in the final remark section****.***

**Next Query to ChatGPT:**

Now I am getting the following error:

HTTPError: HTTP Error 404: Not Found

Please fix it for me.

**Response:**

I apologize for the inconvenience. It seems that the previous URL used to access the dietox dataset is no longer available. However, I can provide an alternative solution by using a local CSV file.

Here's how you can modify the code to load the dietox dataset from a local CSV file:

import pandas as pd

import statsmodels.api as sm

# Load the dietox dataset from a local CSV file

data = pd.read\_csv('path/to/dietox.csv')

# Create a DataFrame with the relevant variables for the model

df = pd.DataFrame(data, columns=['Weight', 'Feed', 'Time'])

# Add a constant column to the DataFrame

df['Intercept'] = 1

# Define the dependent variable

y = df['Weight']

# Define the independent variables

X = df[['Intercept', 'Feed', 'Time']]

# Create the OLS model

model = sm.OLS(y, X)

# Fit the model to the data

results = model.fit()

# Print the summary statistics of the model

print(results.summary())



Make sure to replace 'path/to/dietox.csv' with the actual path to the dietox dataset CSV file on your local system.

By using a local CSV file, you can ensure that the dataset is accessible and avoid the HTTPError 404: Not Found.

***Analysis of the response:***

*Now ChatGPT has just given up, and suggested a solution that beats the whole purpose of loading data from a source other than a local file.*

**Final remark:**

ChatGPT was tasked with a very simple machine learning task of training a linear regression model. The instructions were clearly given. However, it failed in the very first step of any ML process: data collection. Moreover, in one of its proposed solutions it produced a column name ‘weight6’ that does not even exist in the dataset. This shows that ChatGPT is generating text based on what it believes is likely, not based on what is correct and what is incorrect. It does not mean what ChatGPT is generating is entirely useless. We can still use the provided code with some modifications as follows:

from urllib.request import urlopen

import statsmodels.api as sm

import pandas as pd

url = '<https://vincentarelbundock.github.io/Rdatasets/csv/geepack/dietox.csv>'

# Adjustment: ChatGPT provided an incorrect URL, and all we needed to do was change "/boot/" to "/geepack/"

# When asked about the error, ChatGPT was able to assume the URL was no longer available, but wasn't able to provide a correct URL since it's training data predates the change to the URL

data = pd.read\_csv(url)

df = pd.DataFrame(data, columns=['Weight', 'Feed', 'Time'])

# Add a constant column to the DataFrame

df['Intercept'] = 1

# Define the dependent variable

y = df['Weight']

# Define the independent variables

X = df[['Intercept', 'Feed', 'Time']]

# Adjustment: ChatGPT never took care of the missing value which would make the code to fail. We drop missing values in the following two lines.

X.dropna(inplace=True)

y = y[X.index]

# Create the OLS model

model = sm.OLS(y, X)

# Fit the model to the data

results = model.fit()

# Print the summary statistics of the model

print(results.summary())

The above code snippet should read the data from an online source, do data preprocessing by dropping missing values, and train an OLS model successfully.