Supplementary Material

Data analysis using openAI: manually double checking their calculations using Kaggle Notebooks

Jorge Guerra Pires[1]*

Abstract: A good question is whether you should trust or not code interpreter. One solution in case one wants to be sure, it is imperative to avoid errors, is running the calculations yourself. However, there is no evidence this tool is not reliable, it is just in case one wants to make sure, does not trust AIs. On this Supplementary Material, I am going to show one way to verify the calculations from coder interpreter, if you feel the need to do so.

: openAI — data science — statistical inference — python — open source — diabetes

[1] *Founder at IdeaCoding Lab and JovemPesquisador.com*

DOI: http://dx.doi.org/10.22456/2175-2745.XXXX • **Received:** dd/mm/yyyy • **Accepted:** dd/mm/yyyy

CC BY-NC-ND 4.0 - This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License.

1. Understanding apparent temperature vs. temperature

I am not going to validate all the calculations done with coder interpreter. The process is the same for the remaining cases I have considered. The reader is invited to validate if they feel it is necessary. The validation should be done just when there are evidences of errors from the coder interpreter. Or the data scientist suspect something is wrong. It sounds senseless to validate all the calculations, just out of paranoia.

1.1 Details

A good question is whether you should trust or not coder interpreter. One solution is running the calculations yourself. On this Supplementary Material, I am going to show one way to verify the calculations from coder interpreter, if you feel the need to do so.

New instructions:

> Attached is a set of observations of the Apparent Temperature (C) and the Temperature (C) in CSV, also other measurements.
>
> I want to understand better the relation between Apparent Temperature and Temperature. But I want to do myself the calculations using Kaggle Notebooks.
>
> The dataset is already on Kaggle as a public dataset. I want to run manually all the calculations, as you would do. Please, use the same libraries you would have used in your current environment. Follow the same steps you would have followed. Give me the codes in Python, and instructions, I will run them on Kaggle Notebook.
>
> Your jobs is providing codes and instruction. Do not run yourself the analysis [*gpt-3.5-turbo-1106* ignores this instruction, and run the simulations anyway].

It was necessary to change the instructions, as so the assistant would focus on the codes for the Kaggle Notebook. It was also necessary to use *gpt-4-0125-preview* , the defaul I was using did not provide the code. It would run the simulations instead, even when asked not to run. I have used previously *gpt-3.5-turbo-1106* , it is cheaper and faster. But, it is not good at ”hearing instructions”.

Kaggle Notebooks were chosen because: i) they run in Python, the same language from coder interperter; ii) the datasets were taken from this platform, thus, the dataset uploads are straighfoward. Nevertheless, one can do the calculations on any Python notebook.

The dataset was provided as so that the assistant would have more information on the case. However, I believe, a description would also work.

1.2 Conversation

> User: I want to understand better the relation between Apparent Temperature and Temperature.
>
> **Whether bot v2**
>
> Absolutely, understanding the relationship between Apparent Temperature and Temperature can be quite insightful. Let’s approach this with a plan to utilize Python for our analysis.

> You want to perform your own calculations using Kaggle Notebooks, so I'll guide you step-by-step and provide you with the necessary Python code, including explanations of what each part does. You can then execute these instructions in a Kaggle Notebook. We will use common Python libraries for data manipulation and visualization, such as pandas and matplotlib.
>
> **Whether bot v2**
>
> **Step 1: Import Libraries**
>
> First, you'll need to import the necessary libraries. Insert the following Python code at the beginning of your notebook:
>
> ```
> import pandas as pd
> import matplotlib.pyplot as plt
> import seaborn as sns
> import numpy as np
> from sklearn.model_selection
> import train_test_split
> from sklearn.linear_model
> import LinearRegression
> from sklearn.metrics
> import mean_squared_error, r2_score
>
> # Set seaborn style for the plots
> sns.set()
> ```
>
> **Whether bot v2**
>
> **Step 2: Load Your Dataset**
>
> Next, load the dataset you uploaded. The code assumes the dataset is in CSV format and it's named as you uploaded it. Replace `'yourfile.csv'` with the actual file name if different:
>
> ```
> df =
> pd.read_csv('/path/to/your/dataset.csv')
> print(df.head())
> ```

The codes worked, with minor modifications. The full notebook is here on Kaggle.

This approach for validating coder interpreter calculations has a trap: there is no way to know for sure, except if you have access to logs, that the tool is giving you back the same codes and path it followed on the analysis they did all on their environment. One solution is double checking as you do the analysis, therefore, you are in the same thread. The model will have a memory to consult. Should you see discrepancies, you can always talk to the model. Those models are good when you point out directly what you want, rather than expecting a desired behavior from general questioning.