Centre for infrastructure, Sustainable Transportation and Urban Planning

Indian Institute of Science (IISc), Bengaluru

Winter Internship Program 2023

Test for Python Developer Position

The aim of this exam is to test your problem-solving skills and basic understanding of Python. You are encouraged to submit the best possible answer to the question.

Follow the instructions below precisely.

- Plagiarism will result in instant disqualification. You must write your own code.
- To make your submission, use the following Google form: submission link.
  You are allowed to make only one submission for this test. While submitting, you will be asked to upload three documents:
  - A Python file (format: .py) containing all the codes related to the question
  - An output file (format: .txt) from the above code
  - A report (format: pdf) summarising your findings from the question. The report should include essential components such as clearly stated assumptions, informative visualisations, and your findings. Do not copy-paste code in the report.
- If possible, the Python code should be written according to PEP 8 – style guide (reference).
- Your submissions will be evaluated based on the quality of the report and codes. Hence, aim for better visualisations and efficient codes to increase your chances of selection. In case of a tie, the average runtime will be used as a tiebreaker.
- The test commences on 9th November 2023 (10:00 AM). The last date for submission is 11th November 2023 (10:00 AM). Late submissions will not be accepted.

Contact for any clarifications: Dr. Sangram Krishna Nirmale (nirmalek@iisc.ac.in)

All the best.

CiSTUP

IISc, Bengaluru

Write a Python function to calculate the probability of each alternative in a multinomial choice setting using the logistic function, given a set of parameters and independent variables. The function should be generic enough to handle any number of alternatives and independent variables.

In a multinomial logit model, the probability of each alternative is calculated using a logistic function. For each alternative, a deterministic utility (V) is computed based on a linear combination of independent variables and their respective coefficients (β). The probability of each alternative is the exponential of its utility divided by the sum of exponentials of all utilities.

Given Sample Data:

data = {

'X1': [2, 3, 5, 7, 1, 8, 4, 5, 6, 7],

'X2': [1, 5, 3, 8, 2, 7, 5, 9, 4, 2],

'Sero': [0, 0, 0, 0, 0, 0, 0, 0, 0, 0]

}

Deterministic Utilities:

$$V_1 = \beta_{01} + \beta_1 X1 + \beta_2 X2$$

$$V_2 = \beta_{02} + \beta_1 X1 + \beta_2 X2$$

$$V_3 = \beta_{03} + \beta_1 Sero + \beta_2 Sero$$

Probabilities to Compute:

$$P_1 = \frac{exp(V_1)}{exp(V_1) + exp(V_2) + exp(V_3)}$$

$$P_2 = \frac{\exp(V_2)}{\exp(V_1) + \exp(V_2) + \exp(V_3)}$$

$$P_3 = \frac{exp(V_3)}{exp(V_1) + exp(V_2) + exp(V_3)}$$

Parameters:

$$\beta_{01} = 0.1, \beta_1 = 0.5, \beta_2 = 0.5, \beta_{02} = 1, \beta_{03} = 0$$

Tasks

Write a Python function called 'calculate_probabilities' that takes the following inputs:

- Parameters: A dictionary containing the β coefficients.

- Data: A dictionary containing the independent variables (X1, X2, Sero, etc.).

- Utilities: A list of functions that define the deterministic utilities for each alternative based on the given parameters and data.

Your function should output a new dictionary with keys representing each alternative and values as lists containing the calculated probabilities for each data point. Save this output in .txt file format.

Ensure your code is well-commented to explain the logic used at each step.

Bonus: Include error handling for possible input errors such as mismatched dimensions between parameters and data points.

Evaluation Criteria:

- Correctness of the logistic function implementation.
- Ability to handle a dynamic number of alternatives and independent variables.
- Code readability and use of comments.
- Proper error handling.