

UCS654 - Predictive Analytics Using Statistics

Assignment04 - Topsis

General Instructions – Must Read

- **Submission Due Date:** 27 Feb 2022 | 23:59:59
- **Bonus Date [Get addition 20% marks]:** 24 Feb 2022 | 07:59:59
- **Marks:** 10 (Ten)
- **Submission Link:** [Click Here](#)
- **[Learn] Learn Topsis:** [Link-1](#) [Link-2](#)
- **[Learn] How to publish Python package:** [Link-1](#) [Link-2](#) [Link-3](#)
- Input data files are available in “**Input files for Assignment04**” folder
- **Submission Guidelines:**
 - **[Required]** Program 1: One python (.py) file
 - **[Required]** Program 2: Web link of python package for Topsis on pypi.org
 - **[Optional]** Program 3: WebApp link for Topsis
- **Note:**
 - Multiple submissions are allowed, but **latest submission** will be considered for the evaluation.
 - Submission link will open all the time, but only 50% marks will be awarded if you fail to submit with in the due date. No excuse will be consider for the submission.
 - **Zero marks** will be awarded for plagiarized code or result.

Program 1 [Required]: Develop a command line python program to implement the Topsis.

Input File*							Output File*							
Fund Name	P1	P2	P3	P4	P5		Fund Name	P1	P2	P3	P4	P5	Topsis Score	Rank
M1	0.67	0.45	6.5	42.6	12.56		M1	0.67	0.45	6.5	42.6	12.56	20.58	2
M2	0.6	0.36	3.6	53.3	14.47		M2	0.6	0.36	3.6	53.3	14.47	40.83	4
M3	0.82	0.67	3.8	63.1	17.1		M3	0.82	0.67	3.8	63.1	17.1	30.07	3
M4	0.6	0.36	3.5	69.2	18.42		M4	0.6	0.36	3.5	69.2	18.42	50.22	5
M5	0.76	0.58	4.8	43	12.29		M5	0.76	0.58	4.8	43	12.29	10.41	1
M6	0.69	0.48	6.6	48.7	14.12		M6	0.69	0.48	6.6	48.7	14.12	80.51	8
M7	0.79	0.62	4.8	59.2	16.35		M7	0.79	0.62	4.8	59.2	16.35	70.74	7
M8	0.84	0.71	6.5	34.5	10.64		M8	0.84	0.71	6.5	34.5	10.64	60.33	6

* Your input-data and output-result may be different.

1.1 Learn the mathematics of Topsis: [Link-1](#) [Link-2](#)

1.2 Implement the Topsis in python. File name must be <Rollnumber>.py | **Example: 1015579.py**

1.3 Input/Output Files:

- **Input File**
 - Available in “**Input files for Assignment04**” folder
 - First **convert and rename** “data.xlsx” file to <RollNumber>-data.csv | **Example: 101556-data.csv**
 - Input file contain three or more columns
 - First column is the object/variable name (e.g. M1, M2, M3, M4.....)

- From 2nd to last columns contain **numeric values only**
- **Output Files**
 - **Result file** contains all the columns of input file and two additional columns having **Topsis Score and Rank**
 - **Output File Name** must be <RollNumber>-result.csv | **Example: 101556-result.csv**

1.4 Run the program through command line as:

Usages: python <program.py> <InputDataFile> <Weights> <Impacts> <ResultFileName>

Example: python 101556.py 101556-data.csv "1,1,1,2" "+,+,-,+" 101556-result.csv

1.5 Your program must check for:

- Correct number of parameters (inputFileName, Weights, Impacts, resultFileName).
- Show the appropriate message for wrong inputs.
- Handling of "File not Found" exception
- Input file must contain three or more columns.
- From 2nd to last columns must contain numeric values only (Handling of non-numeric values)
- Number of weights, number of impacts and number of columns (from 2nd to last columns) must be same.
- Impacts must be either +ve or -ve.
- Impacts and weights must be separated by ',' (comma).

Program 2 [Required]: Develop a python package and upload it to the **pypi.org**

- **Naming convention** for the package – "Topsis-FirstName-RollNumber"
 - Example: **Topsis-Shyam-10155792**
- [Link-1](#) [Link-2](#) [Link-3](#) to learn "How to build and upload your python package to PyPi".
- Learn how to create python package using Youtube (or any other available resources).
- User Manual must be provided
- Test the package by installing it and run it through command line.
- Make any other assumption if required

Program 3 [Optional]: Develop a web service for Topsis.

File Name	<input type="text" value="Browse File...."/>
Weights	<input type="text" value="1,1,1,1"/>
Impacts	<input "="" type="text" value="+,+,-,+"/>
Email Id	<input type="text" value="psrana@gmail.com"/>
<input type="button" value="Submit"/>	

- User should provide input file, weights, impacts and email id.
- User should get the result file through email.
- Number of weights must be equal to number of impacts
- Impacts must be either +ve or -ve.
- Impacts and weights must be separated by ',' (comma).
- Email id must be correct