AHP

AHP stands for Analytic Hierarchy Process, and it is a decision-making technique developed by Thomas L. Saaty in the 1970s. AHP is a structured methodology that helps individuals and groups make complex decisions by breaking them down into a hierarchical structure.

**1.1 Brief overview of the key concepts in AHP:**

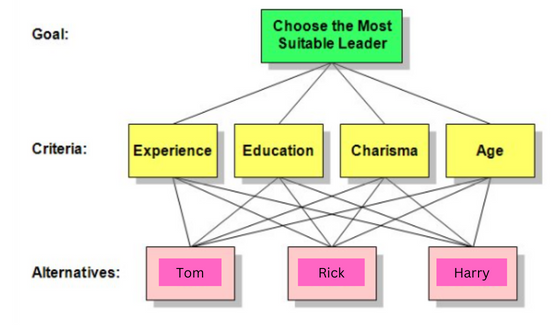
* Hierarchy: AHP involves organizing the decision criteria and alternatives into a hierarchical structure. At the top level is the overall goal or objective, followed by a set of criteria, and further sub-criteria until reaching the most detailed level of alternatives.
* PairwiseComparisons: A fundamental aspect of AHP is the process of pairwise comparisons. Decision-makers assess the relative importance of criteria and alternatives by comparing them in pairs. This is done using a scale of preference, often expressed as numerical values.​
* Consistency Checks: AHP includes a consistency check to ensure the reliability of the pairwise comparisons. If the comparisons are inconsistent, decision-makers may need to revisit their assessments to ensure more accurate and reliable results.
* Calculating Weights: Once we have advanced building a justified comparison matrix between criteria and alternatives, we can figure out their weight to cast their importance with respect to each other.​
* Synthesis and Decision Making: Once the weights are determined, AHP synthesizes the information to provide a clear picture of the preferred alternatives. It helps decision-makers make informed choices by considering both quantitative and qualitative factors in a systematic manner.

**1.2 Application:**

AHP is widely used in various fields, including business, engineering, project management, and other areas where complex decisions need to be made. It provides a structured and transparent approach to decision-making, allowing stakeholders to understand and justify the choices made in a systematic way.

**1.3 A Practical Example: Selecting a new CEO of traditional, big company**

A company want to select a new CEO. There are three valuable candidates - Tom, Rick, Harry. According to the AHP, the first step is to define a hierarchy of sub-criteria that we can use to compare the options. In this case, the board decided that the four main criteria should be: Experience, Education, Charisma, and Age.



Having defined the criteria, we are going to create Weight Comparison Matrix, for each criterion. The WCM is a square NxN matrix, where N is the number of options that we have. The values of this matrix should be filled by comparing the column number option with the line number option in the given criterion, and these values should be defined using the fundamental scale for pairwise comparisons.

|  |  |
| --- | --- |
| Intensity of Importance | Definition |
| 1 | Equal importance |
| 3 | Moderate importance |
| 5 | Strong importance |
| 7 | Very strong importance |
| 9 | Extreme importance |

Fundamental scale of Pairwise Comparisons

After filling the matrix with values, performing –

Inconsistency check:

Checking whether the given matrix is consistency or not, i.e., value should not exceed more than 0.1.

Normalizations:-

Geometric Mean:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Experience** | Tom | Rick | Harry | Priority | | Tom | 1 | ¼ | 4 | 0.217 | | Rick | 4 | 1 | 9 | 0.717 | | Harry | ¼ | 1/9 | 1 | 0.066 | | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Education** | Tom | Rick | Harry | Priority | | Tom | 1 | 3 | 1/5 | 0.188 | | Rick | 1/3 | 1 | 1/7 | 0.081 | | Harry | 5 | 7 | 1 | 0.731 | |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Charisma** | Tom | Rick | Harry | Priority | | Tom | 1 | 5 | 9 | 0.743 | | Rick | 1/5 | 1 | 4 | 0.194 | | Harry | 1/9 | 1/4 | 1 | 0.063 | | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Age** | Tom | Rick | Harry | Priority | | Tom | 1 | 1/3 | 5 | 0.265 | | Rick | 3 | 1 | 9 | 0.672 | | Harry | 1/5 | 1/9 | 1 | 0.063 | |

After defining the WCM of each option for each criterion, now calculate the importance between the criteria themselves. Use the same pairwise comparison method, but here a CxC WCM, where C is the number of criteria.

Performing same Inconsistency check, Normalization and finding Geometric mean.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Criteria** | Experience | Education | Charisma | Age | Priority |
| Experience | 1 | 4 | 3 | 7 | 0.547 |
| Education | ¼ | 1 | 1/3 | 3 | 0.127 |
| Charisma | 1/3 | 3 | 1 | 5 | 0.270 |
| Age | 1/7 | 1/3 | 1/5 | 1 | 0.056 |

Inconsistency- 0.044

Now for final step, calculate the criterion weight by multiplying each criterion priority by the corresponding option priority. Then, at last sum all the weighted criteria by option, having a final score for each option over the main goal.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Candidate | Experience  \*0.547 | Education  \*0.127 | Charisma  \*0.270 | Age  \*0.056 | **Goal** |
| Tom | 0.119 | 0.024 | 0.201 | 0.015 | 0.358 |
| Rick | 0.392 | 0.010 | 0.052 | 0.038 | 0.492 |
| Harry | 0.036 | 0.093 | 0.017 | 0.004 | 0.149 |
| Total: | 0.547 | 0.127 | 0.270 | 0.056 | 1.000 |

Based on the board's choice of decision criteria and their judgment about the relative importance of each one, Rick, with a priority of 0.492, is by far the most suitable candidate. Tom, with a priority of 0.358, is second, and Harry, at 0.149, is third.

**1.4 Package Code**

Code for AHP is divided in two parts. First part is the parameter check (also containing the code of consistency check) and second part contains the logic of implementing AHP.

Input Field:



The given inputs are –

* List of criteria names-(criteria\_names)
* List of alternative names-(alternative\_names)
* List of matrices containing (alt. X alt.) comparison for each criterion- (matrix\_per\_criteria)
* Criteria comparison matrix-(criteria\_comparision)

**Note: Every input matrix should be of NumPy array type except first two.**

Parameter check:- (Matrix\_check\_AHP)







The above code contains various types of error checks to ensure the validity and consistency of the data being processed. Here's a list of the types of error checks performed:

1. Type Check:

- Ensures that `criteria\_names` and `alternative\_names` are of type list.

- Ensures that every element in `criteria\_names` and `alternative\_names` is of type string.

- Checks if `matrix\_per\_criteria` and `criteria\_comparison` are of type numpy.ndarray.

- Checks if `inconsistency` is of type float or int.

2. Dimension Check:

- Verifies that the dimension of `matrix\_per\_criteria` is 3D.

- Checks the dimensionality of `criteria\_comparison` to be 1D or 2D

3. Shape Check:

- Verifies the shape of `matrix\_per\_criteria`, ensuring it matches the expected shape.

- Ensures the shape of `criteria\_comparison` matches the expected shape based on the number of criteria.

4. Value Check:

- Checks if the dtype of `matrix\_per\_criteria` and `criteria\_comparison` is either float or int.

- Verifies that every element in `matrix\_per\_criteria` and `criteria\_comparison` is greater than 0.

5. Consistency Check:

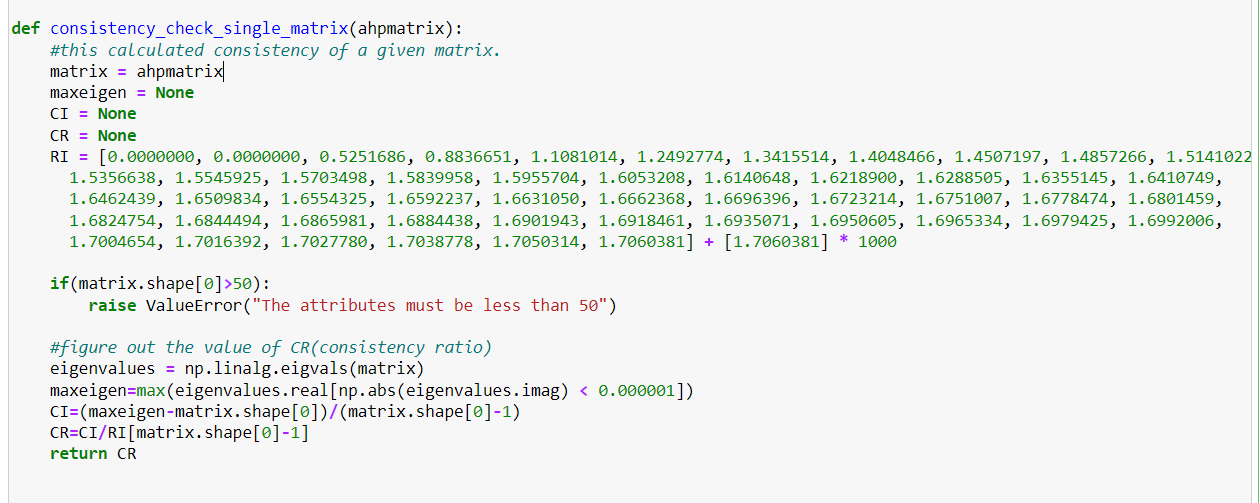
- Ensures that the inconsistency value (`inconsistency`) is a numerical value (integer).

6. Error Reporting:

- Raises ValueError with specific error messages for each type of check that fails.

These error checks collectively ensure that the input data meets the required criteria for further processing, preventing unexpected errors or inconsistencies in the program execution.

Consistency: Below is the code for consistency check for the matrix passed as parameter which is embedded in the Matrix\_check\_AHP class –



Note:

* ‘ahpmatrix’ should be 2d NumPy array with height and width.
* It only works for 50\*50 array at max

AHP code:





The above code defines a class `Ahp` that inherits from another class `Matrix\_check\_AHP`. The `Ahp` class is designed to perform Analytic Hierarchy Process (AHP) computations to determine the most suitable alternative given certain criteria and their comparisons.

Below are the actions of code-

1. Initialization:

- The `\_\_init\_\_` method initializes the `Ahp` object with parameters suchas`criteria\_names`, `alternative\_names`, `matrix\_per\_criteria`, `criteria\_comparison`, `custom\_inconsistency`, `print\_weight\_matrix`, and `print\_rank\_array`.

- It first calls the parent class (`Matrix\_check\_AHP`) constructor using `super().\_\_init\_\_()` to perform necessary checks on the input data.

- It then initializes attributes such as `rank\_array`, `normalized\_unweighted\_matrix`, `weighted\_matrix`, `most\_suitable\_alternative`, `print\_weight\_matrix`, and `print\_rank\_array`.

2. Finding Chart:

- The `find\_chart` method computes the AHP process.

- It computes the normalized unweighted matrix and the weighted matrix.

- It calculates the geometric mean of `matrix\_per\_criteria` along the third axis, representing the criteria.

- It computes the weight array based on `criteria\_comparison`.

- It calculates the rank array by summing the weighted matrix along the rows.

3. \*\*Showing Results\*\*:

- The `show` method displays the computed results.

- It sorts the alternatives based on their rank values and prints them along with their respective ranks.

- It identifies the most suitable alternative based on the highest rank and prints it.

- If `print\_weight\_matrix` is set to `True`, it prints the weighted matrix.

4. Object Creation and Usage:

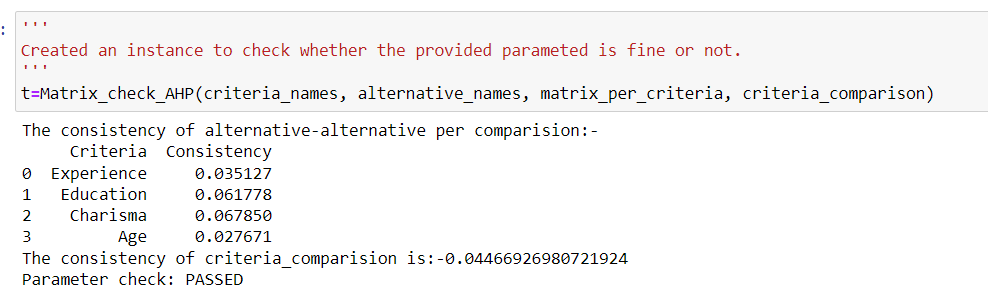
- An instance `a` of the `Ahp` class is created with the specified parameters.

- The `most\_suitable\_alternative` attribute of the `a` object is accessed and printed.

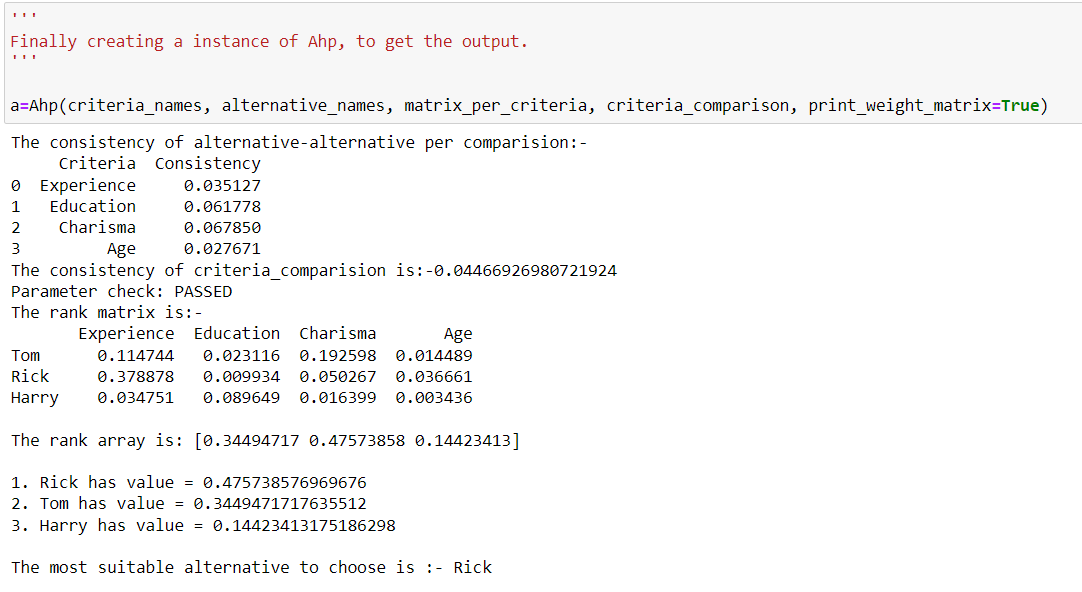
The code effectively encapsulates the AHP process, allowing users to input criteria, alternatives, and comparison matrices to determine the most suitable alternative based on the provided criteria. Additionally, it provides flexibility in displaying intermediate results and final outcomes based on user preferences.

**1.5 Result**

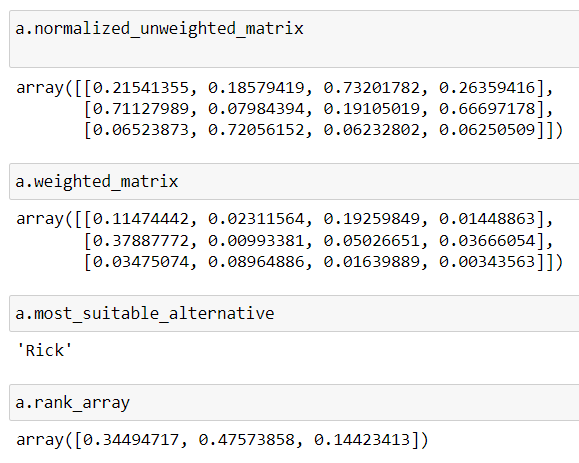
Output of Matrix\_check\_AHP-



Result of AHP-



Other attributes-



A person can check the above attributes to do their own analysis.