Dataset Analysis

The calculation steps using the Simple Additive Weighting (SAW) method:  
1. Determining Alternative (Ai) 2. Determine the criteria to be used as a reference in decision making (Cj) 3. Determine the preference weight or level of importance (W) for each criterion 4. Determine the Match Value of each criterion 5. Make a decision matrix (x) obtained from the suitability rating for each alternative (Ai) with each criterion (Cj). 6. Perform the normalization step of the decision matrix (x) by calculating the value of the normalized performance rating (Rij) from the alternative (Ai) on the criteria (Cj) 7. The result of normalization (Rij) forms a normalized matrix (R) 8. The final result of the preference value (Vi) is obtained from the sum of the normalized matrix row elements (R) with the preference weights (W) corresponding to the matrix column elements (W). With: = rank for each alternative = weighted value of each criterion = normalized performance rating value.

# import csv  
import pandas as pd  
import numpy as np  
import pingouin as pg  
  
# read xls file  
df = pd.read\_excel('quiz\_20221.xlsx')  
df

|  | dosen | ta | nim | kdmk | nmmk | q1 | q2 | q3 | q4 | q5 | q6 | q7 | q8 | q9 | q10 | saran |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | Prof. Dr. Ir EDI NOERSASONGKO, M.Kom | 20221 | A11.2017.10120 | AF201703 | TECHNOPRENEURSHIP | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | NaN |
| 1 | Prof. Dr. Ir EDI NOERSASONGKO, M.Kom | 20221 | A11.2017.10350 | AF201703 | TECHNOPRENEURSHIP | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | NaN |
| 2 | Prof. Dr. Ir EDI NOERSASONGKO, M.Kom | 20221 | A11.2018.11309 | AF201703 | TECHNOPRENEURSHIP | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | NaN |
| 3 | Prof. Dr. Ir EDI NOERSASONGKO, M.Kom | 20221 | A11.2019.11618 | AF201703 | TECHNOPRENEURSHIP | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | ok |
| 4 | Prof. Dr. Ir EDI NOERSASONGKO, M.Kom | 20221 | A11.2019.11622 | AF201703 | TECHNOPRENEURSHIP | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | NaN |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 20092 | DIBYO ADI WIBOWO, S.Si., M.Si | 20221 | A11.2020.80018 | A11.54508 | STRATEGI ALGORITMA | 4 | 3 | 3 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | Tidak ada |
| 20093 | DIBYO ADI WIBOWO, S.Si., M.Si | 20221 | A11.2020.80018 | A11.54812 | METODE NUMERIK | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | Tidak ada |
| 20094 | DEWI PERGIWATI, S.Kom., M.Kom | 20221 | A11.2018.11461 | AF201704 | DASAR DASAR KOMPUTASI | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | NaN |
| 20095 | DEWI PERGIWATI, S.Kom., M.Kom | 20221 | A11.2021.13607 | AF201704 | DASAR DASAR KOMPUTASI | 4 | 4 | 4 | 4 | 4 | 4 | 3 | 3 | 3 | 4 | NaN |
| 20096 | AYU ASHARI, S.S.T, M.Kes | 20221 | A11.2020.13081 | U201701 | DASAR KEWIRAUSAHAAN | 3 | 3 | 3 | 4 | 3 | 3 | 3 | 3 | 3 | 3 | NaN |

df2 = df.copy()  
  
for i in range(len(df2)):  
 df2.loc[i, 'nim'] = df2.loc[i, 'nim'] + str(" - ") + df2.loc[i, 'kdmk']  
  
# remove unnecessary columns  
df2 = df2.drop(['ta','dosen','saran','nmmk','kdmk'], axis=1)  
  
# change column name  
df2 = df2.rename(columns={'nim':'Alternative'})  
  
# insert code column in front of dataframe  
df2.insert(loc=0, column='Code', value=str("A") + df2.index.astype(str))  
  
# remove index column  
df2 = df2.reset\_index(drop=True)  
  
df2

|  | Code | Alternative | q1 | q2 | q3 | q4 | q5 | q6 | q7 | q8 | q9 | q10 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | A0 | A11.2017.10120 - AF201703 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 |
| 1 | A1 | A11.2017.10350 - AF201703 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 |
| 2 | A2 | A11.2018.11309 - AF201703 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 |
| 3 | A3 | A11.2019.11618 - AF201703 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 |
| 4 | A4 | A11.2019.11622 - AF201703 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 20092 | A20092 | A11.2020.80018 - A11.54508 | 4 | 3 | 3 | 4 | 4 | 4 | 4 | 4 | 4 | 4 |
| 20093 | A20093 | A11.2020.80018 - A11.54812 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 |
| 20094 | A20094 | A11.2018.11461 - AF201704 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 |
| 20095 | A20095 | A11.2021.13607 - AF201704 | 4 | 4 | 4 | 4 | 4 | 4 | 3 | 3 | 3 | 4 |
| 20096 | A20096 | A11.2020.13081 - U201701 | 3 | 3 | 3 | 4 | 3 | 3 | 3 | 3 | 3 | 3 |

## Normalize Dataset

df3 = df2.copy()  
  
df3 = df3.drop(['Alternative'], axis=1)  
  
df3

|  | Code | q1 | q2 | q3 | q4 | q5 | q6 | q7 | q8 | q9 | q10 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | A0 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 |
| 1 | A1 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 |
| 2 | A2 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 |
| 3 | A3 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 |
| 4 | A4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 20092 | A20092 | 4 | 3 | 3 | 4 | 4 | 4 | 4 | 4 | 4 | 4 |
| 20093 | A20093 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 |
| 20094 | A20094 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 |
| 20095 | A20095 | 4 | 4 | 4 | 4 | 4 | 4 | 3 | 3 | 3 | 4 |
| 20096 | A20096 | 3 | 3 | 3 | 4 | 3 | 3 | 3 | 3 | 3 | 3 |