Group no. 4

Group Members: Vasanthi Lingamdinne, Niranjan Pawar, Monica Tandel

Assignment 3

In the paper distributed in class on September 11, *Approximating consistency in pairwise comparisons*, by Christopher Kuske, Konrad Kułakowski and Michael Soltys, one of the algorithms proposed is Algorithm 4, which effectively uses a type of binary search to approximate an inconsistent matrix with a consistent one. Implement this algorithm in Python 3. The input to the algorithm should be a text file containing a matrix with rational entries of the form x/y. The operations should be on quotients, i.e., numerator and denominator – the numbers should not be treated as floats, but as pairs of integers. An important questions is: when should the algorithm stop; provide a justification in your writeup. Also prove that if a matrix is consistent, then it is the case for all k that $a_{ip_1}a_{p_2p_3}\cdots a_{p_{k-1}p_k}a_{p_kj} = a_{ij}$.

Ans..

****Note for running the program:**

Input Matrix to the txt file for our python code should be in below format:

3.69981513, 2.6014404, 1.23204583, 0.77434883

2.7539688, 2.55266258, 2.1823636, 3.16786867

3.20347119, 3.4923034, 2.5492294, 1.41945158

1.92904859, 1.78177891, 1.8988549, 2.48834957

1. when should the algorithm stop?

Since we are trying to find a consistent matrix M' = <W> for M of size (nxn).

So, the size of M' should also be (nxn) and size of W is n.

We should run algorithm for all elements of W which is 'n' number of times and repeat this until we stop the algorithm with ceil(max(M)/min(M)) number of times as justified below.

As we are starting our approximation matrix <W> with unit matrix, we have to run this until we reach a closest consistent matrix to M.

2. prove that if a matrix is consistent, then it is the case for all k that $a_{ip_1}a_{p_2p_3}\cdots a_{p_{k-1}p_k}a_{p_kj} = a_{ij}$

Given that matrix M' is consistent, so it is also reciprocal.

Let W = (a_{1k}, $a_{2k,}a_{3k}$, … , a_{nk}) and M' = <W>

As we know that matrix is reciprocal and consistent, therefore a_{ik}/a_{jk} = a_{ij} is true

Then for any i,j entry of <W> it is $w_i/w_j = a_{ik}/a_{jk} = a_{ij.}$

Now, considering:

$$
\begin{aligned}
a_{ip1}\, a_{p1p2}\, a_{p2p3} \ldots\ a_{pk\text{-}1pk}\, a_{pkj} &= (a_{ip1}/a_{p2p1})(a_{p2p3}/a_{p4p3})\,(a_{p3p4}/a_{p5p4}) \ldots (a_{pk-1pk}/a_{jpk}) \\
&= (a_{\cancel{ip1}}/a_{\cancel{p2p1}})(a_{\cancel{p2p3}}/a_{\cancel{p4p3}})\,(a_{\cancel{p3p4}}/a_{\cancel{p5p4}}) \ldots (a_{\cancel{pk-1pk}}/a_{\cancel{jpk}}) \\
&= a_{ij}
\end{aligned}
$$